

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

We, Alexandre Evfimievski, Ramakrishnan Srikant, and Rakesh Agrawal, the Applicants and joint inventors of the above-referenced invention defined by claims 1-24 and disclosed in U.S. Patent Application Serial No. 10/624,069 hereby declare the following:

[0001] The purpose of this declaration is to prove that we conceived the claimed invention prior to the August 2002 date of **Exhibit A**. Exhibit A is a copy of the following published article cited in the March 5, 2008 rejection of claims 1-24 of the present patent application (herein after referred to as Patent Application) under 35 U.S.C. §102(a): Rizvi, et al., "Maintaining Data Privacy in Association Rule Mining," Proceedings of the 28th VLDB Conference, Hong Kong, China, 12 pages, dated August 2002 (hereinafter referred to as Rizvi).

[0002] The following shows that we conceived our invention prior to the August 2002 date of Rizvi, that we were diligent from the date of conception to the date of reduction to

practice and that we were further diligent to the date of the filing of the patent application (herein after referred to as Patent Application), which has a filing date of July 21, 2003.

[0003] Proof of the conception of the claimed invention prior to August 2002 and diligence in reducing the invention to practice and filing the Patent Application is demonstrated by the attached **Exhibit B** in conjunction with **Exhibit A**.

[0004] **Exhibit B** is a copy of the following published paper: Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, "Privacy Preserving Mining of Association Rules," Proc. Of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002, referred to herein as "Privacy Preserving Mining of Association Rules" (July 2002).

[0005] Each of the Applicants of the Patent Application are co-authors on the paper "Privacy Preserving Mining of Association Rules" (July 2002) along with J. Gehrke.

[0006] J. Gehrke was a professor and advisor of A. Evfimievski, during the time period in which the idea for the invention was conceived. Although J. Gehrke is listed as a co-author of "Privacy Preserving Mining of Association Rules" (July 2002), he was not an inventor of the invention defined by claims 1-24 of the Patent Application.

[0007] J. Gehrke has read the Patent Application and has declared that he is not an inventor of the invention defined by claims 1-24 (see **Exhibit C**). We, the Applicants, also acknowledge that J. Gehrke was not an inventor of the invention defined by claims 1-24 of the

Patent Application. Therefore, the portions of “Privacy Preserving Mining of Association Rules” (July 2002), which describe the features of claims 1-24 of the Patent Application, describe the Applicants’ own work and no one else’s.

[0008] “Privacy Preserving Mining of Association Rules” (July 2002) describes the invention defined by claims 1-24. In fact “Privacy Preserving Mining of Association Rules” (July 2002) served as the basis for the specification, drawings and claims of the Patent Application.

[0009] The following is a listing of independent claims 1, 7, 13, and 19 of the Patent Application that define the present invention with reference to the exemplary locations within “Privacy Preserving Mining of Association Rules” (July 2002), wherein the claimed feature is described:

Claim 1: A computer-implemented method of mining association rules over transactions from datasets while maintaining privacy of individual transactions within said datasets through randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see **section 4.1**], said randomizing comprising:

randomly dropping true items from each transaction in said original dataset[see **section 4.1**]; and

randomly inserting false items into each transaction in said original dataset[see **section 4.1**];

collecting said randomized dataset in a database [see section 4.1];
determining support of an association rule in said randomized dataset [see section 4.2];
estimating support of said association rule in said original dataset based on said support
of said association rule in said randomized dataset [see section 4.3]; and
outputting said association rule if said support of said association rule in said original
data set is estimated to be greater than a predetermined minimum [see section 4.5],
wherein, due to said randomizing, privacy breaches of said individual transactions are
controlled [see section 4.4].

Claim 7: A computer-implemented method of mining association rules from
databases while maintaining privacy of individual transactions within said databases through
randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see section 4.1], said
randomizing comprising:

randomly dropping true items from each transaction in said original dataset [see
section 4.1];

randomly inserting false items into each transaction in said original dataset [see
section 4.1];

collecting said randomized dataset in a database [see section 4.1];

mining said database to recover an association rule in said original dataset after said
dropping and inserting processes, wherein said mining comprising [see sections 4.2-4.5]:

determining support for said association rule in said randomized dataset [see
section 4.2];

estimating support of said association rule in said original dataset based on said support of said association rule in said randomized dataset **[see section 4.3]**; and
outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum **[see section 4.5]**,
wherein, due to said randomizing, privacy breaches of said individual transactions are controlled during said mining **[see section 4.4]**.

Claim 13: A computer-implemented method of mining association rules from datasets while maintaining privacy of individual transactions within said datasets through randomization, said method comprising:

creating randomized transactions from an original dataset by **[see section 4.1]**:
randomly dropping true items from each transaction in said original dataset **[see section 4.1]**, and
randomly inserting false items into each transaction in said original dataset **[see section 4.1]**;
creating a randomized dataset by collecting said randomized transactions **[see section 4.1]**;
collecting said randomized dataset in a database **[see section 4.1]**; and
mining said database to recover an association rule in said original dataset after said dropping and inserting processes **[see sections 4.2-4.5]**, wherein said mining comprises:
determining support for said association rule ~~in said~~ in said randomized dataset **[see section 4.2]**;

estimating support of said association rule in said original dataset based on said support for said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said creating of said randomized transactions, privacy breaches of said individual transactions are controlled during said mining [see section 4.4].

Claim 19: A computer program product on a computer-readable medium and tangibly embodying a program of instructions executable by a computer to perform a method of mining association rules from databases while maintaining privacy of individual transactions within said databases through randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see section 4.1], said randomizing comprising:

randomly dropping true items from each transaction in said original dataset [see section 4.1];

randomly inserting false items into each transaction in said original dataset [see section 4.1];

collecting said randomized dataset in a database [see section 4.1]; and

mining said database to recover an association rule in said original dataset after said dropping and inserting processes [see sections 4.2-4.5], wherein said mining comprises:

determining support for said association rule in said randomized dataset [see section 4.2];

estimating support of said association rule in said original dataset based on said support of said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said randomizing, privacy breaches of said individual transactions are controlled during said mining [see section 4.4].

[0010] Furthermore, dependent claims 2-6, 8-12, 14-18 and 20-24 are either explicitly described in “Privacy Preserving Mining of Association Rules” (July 2002) or inferred from details contained therein.

[0011] “Privacy Preserving Mining of Association Rules” (July 2002) clearly predates the August 2002 date of Rizvi. Additionally, at the August 2002 date of Rizvi, the authors of Rizvi had knowledge of the details of the present invention and wrote their paper in light of that knowledge. This is evidenced by the fact that, as mentioned above, the details of the invention as defined by claims 1-24 of the Patent Application are described in “Privacy Preserving Mining of Association Rules” (July 2002) and further by the fact that Rizvi cites “Privacy Preserving Mining of Association Rules” (July 2002), as a reference, at various places throughout the text of the article.

[0012] We were diligent from the date of conception in reducing the invention to practice and in pursuing, preparing, and filing the Patent Application.

[0013] On May 15, 2003, a patent attorney was instructed to prepare a patent application that eventually became the Patent Application. The Patent Application was eventually prepared and filed on July 21, 2003.

[0014] Finally, the above declarations are made according to the best of my/our recollection upon review of the appropriate documents and notes, and I/we hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the Patent Application or any patent issuing thereon. All statements that are made herein of my/our own knowledge are true and all statements that are made herein based on information and belief are believed to be true.

Evfimievski June 30, 2008
Alexandre Evfimievski (Date)

R. A. July 2, 2008
Ramakrishnan Srikant (Date)

Rakesh Agrawal (Date)

Maintaining Data Privacy in Association Rule Mining

Shariq J. Rizvi

Computer Science & Engineering
Indian Institute of Technology
Mumbai 400076, INDIA
rizvi@cse.iitb.ac.in

Jayant R. Haritsa*

Database Systems Lab, SERC
Indian Institute of Science
Bangalore 560012, INDIA
haritsa@dsl.serc.iisc.ernet.in

Abstract

Data mining services require accurate input data for their results to be meaningful, but privacy concerns may influence users to provide spurious information. We investigate here, with respect to mining association rules, whether users can be encouraged to provide correct information by ensuring that the mining process cannot, with any reasonable degree of certainty, violate their privacy. We present a scheme, based on probabilistic distortion of user data, that can simultaneously provide a high degree of privacy to the user and retain a high level of accuracy in the mining results. The performance of the scheme is validated against representative real and synthetic datasets.

1 Introduction

The knowledge models produced through data mining techniques are only as good as the accuracy of their input data. One source of data inaccuracy is when users deliberately provide wrong information. This is especially common with regard to customers who are asked to provide personal information on Web forms to e-commerce service providers. The compulsion for doing so may be the (perhaps well-founded) worry that the requested information may be misused by the service provider to harass the customer. As a case in point, consider a pharmaceutical company that asks clients to disclose the diseases they have suffered from in order to investigate the correlations in their occurrences

— for example, “Adult females with malarial infections are also prone to contract tuberculosis”. While the company may be acquiring the data solely for genuine data mining purposes that would eventually reflect itself in better service to the client, at the same time the client might worry that if her medical records are either inadvertently or deliberately disclosed, it may adversely affect her employment opportunities.

We investigate, in this paper, whether customers can be encouraged to provide correct information by ensuring that the mining process cannot, with any reasonable degree of certainty, violate their privacy. At the same time, we would like the mining process to be as accurate as possible in terms of its results. The difficulty lies in the fact that these two metrics: *privacy* and *accuracy*, are typically contradictory in nature, with the consequence that improving one usually incurs a cost in the other [1]. Therefore, we comprise on the ideal and perhaps infeasible goal of having both complete privacy and complete accuracy through *approximate* solutions that provide practically acceptable values for these metrics. Note further that since the purpose of data mining is essentially to identify statistical *trends*, cent-per-cent accuracy in the mining results is perhaps often not even a required feature.

Our study is carried out in the context of extracting *association rules* from large historical databases, a popular mining process [2] that identifies interesting correlations between database attributes, such as the one described in the pharmaceutical example. For this framework, we present a scheme called **MASK** (Mining Associations with Secrecy Konstraints), that attempts to simultaneously provide a high degree of privacy to the user and retain a high degree of accuracy in the mining results. Our scheme is based on a simple probabilistic distortion of user data, employing random numbers generated from a pre-defined distribution function. It is this distorted information that is eventually supplied to the data miner, along with a description of the distortion procedure. We define a privacy metric and present an analytical formula for evaluating the privacy obtained under this metric by the distortion approach.

*Contact Author

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the VLDB copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Very Large Data Base Endowment. To copy otherwise, or to republish, requires a fee and/or special permission from the Endowment.

**Proceedings of the 28th VLDB Conference,
Hong Kong, China, 2002**

A special feature of our scheme is that the distortion process can be easily implemented at the data source itself, that is, at the *user machine*. This increases the confidence of the user in providing accurate information since she does not have to trust a third-party to carry out the distortion process before the data is acquired by the service provider. Note that some of the other privacy techniques suggested in the literature, such as swapping values between records [7], do not support this feature since they require the entire database to be available for their functioning.

As described in detail later in the paper, mining the distorted database can be, apart from being error-prone, significantly *more expensive* in terms of both time and space as compared to mining the true database. We present a variety of optimizations to address these issues.

Finally, the performance of MASK’s mining scheme is validated against representative real and synthetic datasets, with respect to both privacy and accuracy. Our results indicate that there are regions of the distortion parameter space that are conducive to satisfactorily meeting the dual objectives.

1.1 Organization

The remainder of this paper is organized as follows: In Section 2, we describe the privacy framework employed in our study and in Section 3, we quantify the privacy attained under this framework by our distortion method. Then, in Section 4, we present our new MASK algorithm for mining the distorted database. Optimizations to improve the space and time complexity of MASK are described in Section 5. The performance model and the experimental results are highlighted in Sections 6 and 7, respectively. Bounds on the reconstruction errors incurred during the mining process are given in Section 8. Related work on privacy-preserving mining is reviewed in Section 9. Finally, in Section 10, we summarize the conclusions of our study and outline future avenues to explore.

2 Problem Framework

In this section, we describe the framework of the privacy mining problem that we consider here.

2.1 Database Model

We assume that each customer contributes a tuple to the database with the tuple being a fixed-length sequence of 1’s and 0’s. A typical example of such a database is the so-called “market-basket” database [2] wherein the columns represent the items sold by a supermarket, and each row describes, through a sequence of 1’s and 0’s, the purchases made by a particular customer (1 indicates a purchase and 0 indicates no purchase). We also assume that the overall number of 1’s

in the database is significantly smaller than the number of 0’s – this is especially true for market-baskets since each customer typically buys only a small fraction of all the items available in the store. In short, the database is modeled as a *large disk-resident two-dimensional sparse boolean matrix*.

Note that the boolean representation is only logical and that the database tuples may actually be physically stored as “item-lists”, that is, as an ordered list of the identifiers of the items purchased in the transaction. The list representation may appear preferable for the sparse databases that we are considering, since it reduces the space requirement as compared to storing entire bit-vectors. However, because of the fact that we are *distorting* user information, it may be the case that the distorted matrix will not be as sparse as the true database. Therefore, in this paper, we assume that the distorted database is stored as a large collection of bit-vectors.

2.2 Mining Objectives

The goal of the miner is to compute *association rules* on the above database. Denoting the set of transactions in the database by \mathcal{T} and the set of items in the database by \mathcal{I} , an association rule is a (statistical) implication of the form $\mathcal{X} \implies \mathcal{Y}$, where $\mathcal{X}, \mathcal{Y} \subset \mathcal{I}$ and $\mathcal{X} \cap \mathcal{Y} = \phi$. A rule $\mathcal{X} \implies \mathcal{Y}$ is said to have a *support* (or frequency) factor s iff at least $s\%$ of the transactions in \mathcal{T} satisfy $\mathcal{X} \cup \mathcal{Y}$. A rule $\mathcal{X} \implies \mathcal{Y}$ is satisfied in the set of transactions \mathcal{T} with a *confidence* factor c iff at least $c\%$ of the transactions in \mathcal{T} that satisfy \mathcal{X} also satisfy \mathcal{Y} . Both support and confidence are fractions in the interval $[0,1]$. The support is a measure of statistical significance, whereas confidence is a measure of the strength of the rule.

A rule is said to be “interesting” if its support and confidence are greater than user-defined thresholds sup_{min} and con_{min} , respectively, and the objective of the mining process is to find all such interesting rules. It has been shown in [2] that achieving this goal is effectively equivalent to generating all subsets \mathcal{X} of \mathcal{I} that have support greater than sup_{min} – these subsets are called *frequent* itemsets. Therefore, the mining objective is, in essence, to efficiently discover all frequent itemsets that are present in the database.

2.3 Privacy Metric

As mentioned earlier, the mechanism adopted in this paper for achieving privacy is to *distort* the user data before it is subject to the mining process. Accordingly, we measure privacy with regard to the probability with which the user’s distorted entries can be *reconstructed*. While privacy could be measured at the granularity of entire tuples, we consider here the *stronger* requirement of ensuring privacy at the level of *individual* entries in each customer tuple. In short, our privacy

metric is: “With what probability can a given 1 or 0 in the true matrix be reconstructed”?

A related issue here is whether the user would want the *same* level of privacy for both 1’s and 0’s? For many applications, such as the market-basket database, it appears reasonable to expect that customers would want more privacy for their 1’s than for their 0’s, since the 1’s denote specific actions whereas the 0’s are the default options.

3 Quantifying MASK’s Privacy

In this section, we present the distortion procedure used by the MASK scheme and quantify the privacy provided by the procedure, as per the above metric.

3.1 Distortion Procedure

A customer tuple can be considered to be a random vector $\mathbf{X} = \{\mathbf{X}_i\}$, such that $X_i = 0$ or 1. We generate the distorted vector from this customer tuple by computing $\mathbf{Y} = \text{distort}(\mathbf{X})$ where $\mathbf{Y}_i = \mathbf{X}_i \text{ XOR } \bar{r}_i$ and \bar{r}_i is the complement of r_i , a random variable with density function $\mathbf{f}(\mathbf{r}) = \text{bernoulli}(\mathbf{p})$ ($0 \leq \mathbf{p} \leq 1$). That is, r_i takes a value 1 with probability p and 0 with probability $1 - p$.

The net effect of the above computation is that the identity of the i^{th} element in X is kept the same with probability p and is *flipped* with probability $(1 - p)$. All the customer tuples are distorted in this fashion and make up the database supplied to the miner – in effect, the miner receives a *probabilistic function* of the true customer database.

Note that, in principle, it is possible to use different settings of p for distorting different items. That is, to have a *vector* of p settings ranging across the columns of the database. For simplicity, we will assume here that a single p is used for all the items – this choice also has useful implementation implications, as described later in Section 5.

We now move on to quantifying the privacy obtained by the above distortion procedure. In the following analysis we first consider the extreme case, where the user wants to have maximum privacy for 1’s but is completely unconcerned about the 0’s. After that, we derive the general privacy equations where the customer, though more conservative about 1’s, requires a degree of privacy for the 0’s too.

A caveat: Our privacy estimates do not take into account the fact that there may be a reduction in privacy, as pointed out recently in [1, 8], when the mining output (i.e., the association rules) is used to re-interrogate the distorted database – we plan to investigate this issue in our future work.

3.2 Reconstruction Probability of a 1

Let s_i be the true support of item i , normalized to the number of tuples in the database. This means that the

probability that a random customer \mathcal{C} bought this i^{th} item is s_i . We now have to evaluate the probability that given that \mathcal{C} indeed did buy item i , her original ‘1’ can be reconstructed from the distorted entry. Denoting the original entry as X_i and the distorted entry as Y_i , the probability of correct reconstruction is given by:

$$\begin{aligned} \mathcal{R}_1(p, s_i) = & \\ & P_r\{Y_i = 1|X_i = 1\} \times P_r\{X_i = 1|Y_i = 1\} \\ & + \\ & P_r\{Y_i = 0|X_i = 1\} \times P_r\{X_i = 1|Y_i = 0\} \end{aligned}$$

This expression captures the “round-trip” of going from the true database to the distorted database and then returning to guess the contents of the true database. It can be simplified to

$$\begin{aligned} \mathcal{R}_1(p, s_i) = & \\ & p \times P_r\{X_i = 1|Y_i = 1\} \\ & + \\ & (1 - p) \times P_r\{X_i = 1|Y_i = 0\} \end{aligned}$$

But, we know that

$$\begin{aligned} P_r\{X_i = 1|Y_i = 1\} &= \frac{P_r\{X_i=1 \cap Y_i=1\}}{P_r\{Y_i=1\}} \\ &= \frac{P_r\{X_i=1\} \times P_r\{Y_i=1|X_i=1\}}{P_r\{Y_i=1\}} \\ &= \frac{s_i \times p}{P_r\{X_i=1\} \times P_r\{Y_i=1|X_i=1\} + P_r\{X_i=0\} \times P_r\{Y_i=1|X_i=0\}} \\ &= \frac{s_i \times p}{s_i \times p + (1-s_i) \times (1-p)} \end{aligned}$$

Similarly,

$$P_r\{X_i = 1|Y_i = 0\} = \frac{s_i \times (1-p)}{s_i \times (1-p) + (1-s_i) \times p}$$

Putting it all together, we obtain

$$\mathcal{R}_1(p, s_i) = \frac{s_i \times p^2}{s_i \times p + (1-s_i) \times (1-p)} + \frac{s_i \times (1-p)^2}{s_i \times (1-p) + (1-s_i) \times p}$$

The above expression reflects the reconstruction probability of a ‘1’ in a random item i . To find a total measure of reconstruction, we range across all items:

$$\mathcal{R}_1(p) = \frac{\sum_i s_i \mathcal{R}_1(p, s_i)}{\sum_i s_i} \quad (1)$$

The above expression is *minimized* when all the items in the database have the same support, and increases with the variance in the supports across items. As discussed later in Section 7, with the appropriate choice of p , this increase is marginal for the market-basket type of datasets that we consider here. Therefore, as a first-level approximation, we replace the item-specific supports in the above equation by s_0 , the average support of an item in the database. With this, the reconstruction probability simplifies to

$$\mathcal{R}_1(p) = \frac{s_0 \times p^2}{s_0 \times p + (1-s_0) \times (1-p)} + \frac{s_0 \times (1-p)^2}{s_0 \times (1-p) + (1-s_0) \times p} \quad (2)$$

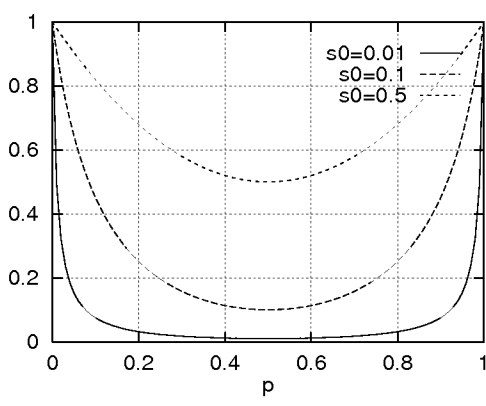


Figure 1: Reconstruction Probability $\mathcal{R}_1(p)$

We can plot $\mathcal{R}_1(p)$ as a function of p for different values of s_0 , as shown in Figure 1. We observe here that:

1. The reconstruction probability is high at the extremes and lowest at the center (i.e $p = 0.5$). This is to be intuitively expected since setting $p = 0.5$ imparts the maximum randomness to the distorted values.
2. The curves are *symmetric* around $p = 0.5$. The implication of this is that there is no difference, reconstruction-wise, between choosing a value p or its counterpart $1 - p$. This may appear surprising at first glance since a matrix that is distorted with $p = 0.1$ would tend to “look” very different with regard to the true matrix, as compared to the same matrix distorted with $p = 0.9$. However, recall that the data miner is also provided with a description of the distortion procedure, that is, he knows the value of p . In this situation, mere differences in appearance do not result in any additional privacy. A practical use of this feature, however, is in *psychological* terms: Distorting with a low value of p as opposed to its high-valued complement, might be more comforting to the user since at least visually it will appear to be considerably different from the true information that she had supplied.
3. Although the minimum always is at $p = 0.5$, the curves become flatter as the average support of items decreases. For a typical market-basket type database with an average transaction length of 10 and the number of items being 1000, the average support is 0.01, which corresponds to the lowest curve in Figure 1.

In the above derivation, it may appear unintuitive that the reconstruction probability depends on the *support* of items. The reason for this is the following: We are considering the possibility of reconstruction of the true value of an entry given the distorted entry. If the data

miner gets a ‘1’ (or a ‘0’) for a particular entry in the distorted database, the probability that it came from a ‘1’ in the true database not only depends on p but also on the distribution of 1’s and 0’s in the true database.

Yet another issue is that we have used the *true* supports of items in the derivation, but these values are not known to the data miner. Therefore, it may appear that we are overestimating the reconstruction probability. However, the point is that since the ultimate goal is to be able to mine the distorted database correctly, we make the conservative assumption that the miner will be able to derive reasonably accurate item supports, implying that he *does* have access to the s_i values.

3.3 The General Reconstruction Equation

We now move on to deriving the relationship between p and the reconstruction probability for the general case where the customer may wish to protect both her 1’s and 0’s.

Analogous to the manner in which we computed $\mathcal{R}_1(p)$ above, we can derive the probability with which a ‘0’ can be reconstructed as:

$$\mathcal{R}_0(p, s_i) =$$

$$\begin{aligned} &P_r\{Y_i = 1|X_i = 0\} \times P_r\{X_i = 0|Y_i = 1\} \\ &+ \\ &P_r\{Y_i = 0|X_i = 0\} \times P_r\{X_i = 0|Y_i = 0\} \end{aligned}$$

leading to

$$\mathcal{R}_0(p) = \frac{(1-s_0) \times p^2}{(1-s_0) \times p + s_0 \times (1-p)} + \frac{(1-s_0) \times (1-p)^2}{s_0 \times p + (1-s_0) \times (1-p)}$$

Our aim is to minimize a weighted average of $\mathcal{R}_1(p)$ and $\mathcal{R}_0(p)$. This corresponds to minimizing the probability of reconstruction of both 1’s and 0’s. The weight denotes the preference which the privacy of 1’s has over that of 0’s. The total reconstruction probability, $\mathcal{R}(p)$, is then given as

$$\mathcal{R}(p) = a\mathcal{R}_1(p) + (1-a)\mathcal{R}_0(p) \quad (3)$$

where a is the weight given to 1’s over 0’s. Note that the a setting must incorporate the fact that the number of 0’s in the database is more than that of 1’s. So, for example, if we set $a = 0.5$ for a database that has $s_0 = 0.01$, we are indicating that the privacy of 1’s is 99 times more critical than that of 0’s (as the number of 0’s is 99 times more than that of 1’s).

3.4 Privacy Measure

Armed with the ability to compute the reconstruction probability, we now simply define user privacy as the following percentage:

$$\mathcal{P}(p) = (1 - \mathcal{R}(p)) * 100. \quad (4)$$

p	Privacy Attained
0.5	89%
0.7	88%
0.8	87%
0.9	83%
0.95	76%
1	0%

Table 1: Privacy attained with $s_0 = 0.01$ and $a = 0.9$

That is, when the reconstruction probability is 0, the privacy is 100%, whereas it is 0 if the $\mathcal{R}(p) = 1$. In Figure 2, we plot this user privacy as a function of p for $s_0 = 0.01$ with different values of a . Note that for a given value of s_0 , the shape of the curve is fixed, and it is only the value of a that decides the absolute value of the attained privacy.

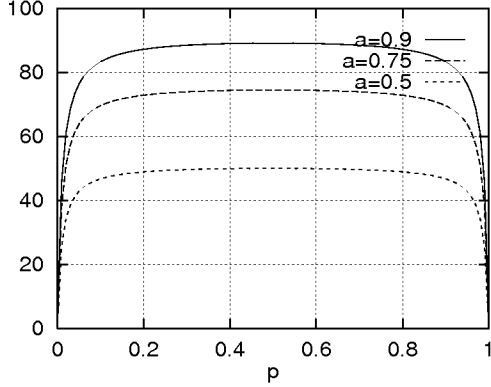


Figure 2: Privacy $\mathcal{P}(p)$ attained for $s_0 = 0.01$

Further, note that the curves have the “knee-points” at $p = 0.1$ and $p = 0.9$ and that for $a = 0.9$, the privacy is almost constant at a high value of around 85% in this large range (0.1 to 0.9) of distortion probabilities – the explicit values are shown in Table 1. This result is very encouraging since it means that we now have considerable flexibility in choosing the p value – in particular, we can choose it in a manner that will *minimize the error* in the subsequent mining process.

4 Mining the Distorted Database

Having established the privacy obtained from our distortion procedure, we now move on to presenting MASK’s technique for estimating the true (accurate) supports of itemsets from a distorted database. Later, in Section 5, we present a variety of optimizations that help to speed up the estimation process. Finally, in Section 7, we evaluate the quality of these estimations.

In the following discussion, we first show how to estimate the supports of 1-itemsets (i.e. singletons) and then present the general n -itemset support estimation procedure. In this derivation, it is important to keep in mind that the miner is provided with both the dis-

torted matrix as well as the distortion procedure, that is, he *knows* the value of p that was used in distorting the true matrix.

4.1 Estimating Singleton Supports

We denote the original true matrix by T and the distorted matrix, obtained with a distortion probability of p , as D . Now consider a random individual item i . Let c_1^T and c_0^T represent the number of 1’s and 0’s, respectively, in the i column of T , while c_1^D and c_0^D represent the number of 1’s and 0’s, respectively, in the i column of D . With this notation, we estimate the support of i in T using the following equation:

$$\mathbf{C}^T = \mathbf{M}^{-1} \mathbf{C}^D \quad (5)$$

where

$$\mathbf{M} = \begin{bmatrix} p & 1-p \\ 1-p & p \end{bmatrix} \quad \mathbf{C}^D = \begin{bmatrix} c_1^D \\ c_0^D \end{bmatrix} \quad \mathbf{C}^T = \begin{bmatrix} c_1^T \\ c_0^T \end{bmatrix}$$

The \mathbf{M} matrix in the above equation incorporates the observation that by our method of distortion, if a column had n 1’s in T , these 1’s will generate approximately pn 1’s and $(1-p)n$ 0’s for the same column in D . Similarly for the 0’s of this column in T . Therefore, given c_1^D and c_0^D , it is possible to estimate the value of c_1^T , that is, the true support of item i .

Note also that the equation rules out the possibility of using $p = 0.5$ because at this value the matrix becomes singular and is not invertible. Intuitively, this happens because at this value of p , the matrix \mathbf{C}^D does not carry sufficient information to be able to reconstruct the values of c_1^T and c_0^T .

4.2 Estimating n -itemset Supports

It is easy to extend Equation 5, which is applicable to individual items, to compute the support for an arbitrary n -itemset. For this general case, we define the matrices as:

$$\mathbf{C}^D = \begin{bmatrix} c_{2^n-1}^D \\ \vdots \\ c_1^D \\ c_0^D \end{bmatrix} \quad \mathbf{C}^T = \begin{bmatrix} c_{2^n-1}^T \\ \vdots \\ c_1^T \\ c_0^T \end{bmatrix}$$

Here c_k^T should be interpreted as the count of the tuples in T that have the binary form of k (in n digits) for the given itemset (that is, for a 2-itemset, c_2^T refers to the count of 10’s in the columns of T corresponding to that itemset, c_3^T to the count of 11’s, and so on). Similarly, c_k^D is defined for the distorted matrix D .

Finally, the matrix \mathbf{M} is defined as:

$$m_{i,j} = \text{The probability that a tuple of the form corresponding to } c_j^T \text{ in } T \text{ goes to a tuple of the form corresponding to } c_i^D \text{ in } D$$

For example, $m_{1,2}$ for a 2-itemset is the probability that a 10 tuple distorts to a 01 tuple. Accordingly, $m_{1,2} = (1 - p)(1 - p)$. The basis for this formulation lies in the fact that in our distortion procedure, the component columns of an n -itemset are distorted *independently*. Therefore, we can use the product of the probability terms.

At first glance, it might seem that the above mining process may need an *exponential* number of counters (2^n counters for an n -itemset), making the process infeasible in practice. But, in fact, this is not really so if the same value of p is used for all items. This is because it results in the matrix \mathbf{M} having interesting symmetry properties that are reflected in \mathbf{M}^{-1} also. In particular, we show in Section 5 how a *linear* number of counters (specifically, $n + 1$ counters for an n -itemset) are now sufficient to complete the mining process.

4.3 The Full Mining Process

The above equations help us to estimate the value of $c_{2^n-1}^T$ for an n -itemset by using the values of c_i^D , $0 \leq i \leq 2^n - 1$. But, first we need to compute the c_i^D values themselves. For this purpose, in principle we could use, after the modifications described below, any of the numerous association rule mining algorithms proposed in the literature (e.g. [3, 13, 9, 16]). With a view to simplicity, we have currently implemented our system based on the classical *Apriori* algorithm [3]. Apriori is a multi-pass algorithm wherein the i^{th} pass computes the frequent i -itemsets by counting all candidate itemsets associated with the pass and, after each pass, the *AprioriGen* algorithm is used for generating the candidate itemsets for the next pass. In our approach too, the i^{th} pass identifies large i -itemsets, and the AprioriGen algorithm is used for generating the candidate itemsets for the next pass.

A critical difference between our approach and that of Apriori, however, is the following: Consider that we are counting, say, 2-itemsets. Here, Apriori only needs to keep track, for each candidate 2-itemset, of the number of tuples in which there was a ‘1’ for both the items appearing in the itemset. That is, it needs to count only the ‘11’s. But, in our case, we need to keep track of *all* combinations: 00, 01, 10, and 11. This is a direct fallout of the fact that we have distorted the true matrix, and an original ‘11’ can now potentially be any of the four combinations. We describe in Section 5 a simple optimization that can significantly reduce the total amount of counting.

Another important point to note here is that the equation to compute the true supports needs to be evaluated only at the *end of every pass* over the distorted matrix, and not on a tuple-by-tuple basis. Further, if the same p value is used for all columns, the matrix \mathbf{M} is identical for *all* candidate n -itemsets and therefore has to be generated *only once* at the end of

each pass. Finally, note that the size of the square matrix is $\mathbf{O}(2^n)$ for candidate n -itemsets.

5 MASK Mining Optimizations

We now describe a set of optimizations to improve the efficiency of the mining process described in the previous section.

5.1 Linear Number of Counters

We first consider how to reduce the number of counters required for each itemset. At the end of the n^{th} pass, we generate a square matrix of size 2^n which depends only on p . We then invert it and multiply the resulting matrix with the counts of the 2^n components of every n -itemset. In effect, the reconstructed support is a *weighted sum* of the counts of all 2^n components in the distorted database. A close observation will reveal, however, that these 2^n weights have *only* $n + 1$ *distinct* weights in them.

For example, for a 2-itemset, we have the estimated reconstructed support value to be

$$s_{est} = a_1 C_{00}^D + a_2 C_{01}^D + a_3 C_{10}^D + a_4 C_{11}^D$$

where C_{xy}^D is the count of xy tuples in the distorted database, and the a_i are the associated weights. Here, the weights a_2 and a_3 will be equal because the probability that a ‘11’ distorts to a ‘10’ is equal to the probability that a ‘11’ distorts to a ‘01’ (both are $p(1 - p)$). Hence, the *reverse* component weights are also equal. Therefore, overall, we need to maintain only 3 counters – one each for 00’s and 11’s, and a third which is common for 01’s and 10’s.

The above observation can be generalized to an n -itemset. For the 2^n counts we have merely $n + 1$ distinct weights: One for ‘00...0’, one for ‘11...1’ and one each for components that have the same number of 1’s (or equivalently 0’s) in them. For example, for a 3-itemset, only 4 counters are required: one each for 000 and 111, one for the triplet (001,010,100) and the fourth for the triplet (011,101,110).

5.2 Reducing Amount of Counting

Apart from reducing the *number* of counters, we can also achieve some reductions in the *amount* of counting by making use of simple algebraic properties.

For example, consider 2-itemsets: The counting of only 11’s takes $\mathbf{O}(tlen^2)$ operations, where $tlen$ is the transaction length. However, counting all 4 components (00,01,10,11), which we have to do for the distorted matrix, takes $\mathbf{O}(|\mathbf{Candidates}|)$ operations. Since the number of candidate 2-itemsets is typically very large, the latter will take much more time than the ordinary mining process.

We can optimize the above by using the observation that $c_3^D + c_2^D + c_1^D + c_0^D$ must be equal to the database

cardinality, *dbsize*. This means that we can choose to *not count* one of these components, say ‘00’s, since it is derivable from the other counts. In conjunction with the counting process described below, this optimization can result in considerable savings.

Our counting process examines the (distorted) customer tuples one by one. For the current customer tuple, the purchase vector of 1’s and 0’s is converted into an *item-list* that contains only the identifiers of the items that have a ‘1’ in the transaction. From this list, the identifiers of all the (previously estimated) infrequent 1-itemsets are removed. The next stage is to create a *complement-list* that consists of all the (previously estimated) frequent 1-itemsets that *do not appear* in this transaction. Let the item-list and the complement-list be of lengths m_1 and m_2 , respectively, (with $m_1 + m_2 = |F_1|$, the total number of frequent 1-itemsets). If we now restrict our counting to the 11’s, 01’s and 10’s, it will take $O(m_1^2) + O(m_1 m_2)$ operations. For high settings of the p parameter, the value of m_1 will be rather small and the transaction will have a large number of ‘00’ pairs. The approach of not counting the ‘00’s can therefore result in significant savings in such situations. In fact, we observed in our experiments (described in Section 7) that for $p = 0.9$, the execution time for Pass 2 reduced by a factor of *four* due to this optimization.

6 Performance Framework

The privacy of the MASK scheme was analytically evaluated in Section 3. We now move on to evaluating its accuracy with regard to the mining results that it derives from the distorted database.

Since MASK is a *probabilistic* approach, fundamentally we cannot expect the reconstructed support values to co-incide exactly with the actual supports. This means that we may have errors in the estimated supports of frequent itemsets with the reported values being either larger or smaller than the actual supports.

Errors in support estimation can have an even more pernicious effect than just wrongly reporting the support of a frequent itemset. They can result in errors in the *identities* of the frequent itemsets. This becomes especially an issue when the sup_{min} setting is such that the support of a number of itemsets lie very close to this threshold value. Typically, this happens for lower values of sup_{min} due to the larger number of frequent itemsets in the database. Such “border-line” itemsets may get wrongly reported as either frequent or rare, based on how the probabilistic evaluation estimates their supports. That is, we can encounter both *false positives* and *false negatives*.

Worse, errors in association rule mining *percolate* through the various passes of the mining process – that is, an error in identifying a 1-itemset correctly has a ripple effect in terms of causing errors in the remainder of the frequent itemset lattice.

To assess the above effects, we evaluate the mining process under two conditions: The first with the sup_{min} value provided by the user, and the second with a marginally *lower* value. The lower value is expected to help reduce the number of false negatives at the risk of increasing the number of false positives, but this is in keeping with the view that it is more important to have coverage as compared to precision. For the experiments described here, we evaluate the impact of a 10% reduction, denoted $r = 10\%$, in the sup_{min} value.

To quantify the errors that we are making, we compare our outputs with those derived from Apriori running on the true database with both the user provided sup_{min} , as well as with the lowered value mentioned above.

6.1 Data Sets

Our evaluations were carried out on two representative databases:

1. A synthetic database generated from the IBM Almaden generator [3]. The synthetic database was created with parameters T10.I4.D1M.N1K (as per the naming convention of [3]), resulting in a million customer tuples with each customer purchasing about ten items on average.
2. A real dataset, BMS-WebView-1 [20], placed in the public domain by Blue Martini Software. This database contains click-stream data from the web site of a (now defunct) legwear and legcare retailer. There are about 60,000 tuples with close to 500 items in the schema. In order to ensure that our results were applicable to large disk-resident databases, we scaled this database by a factor of ten, resulting in approximately 0.6 million tuples.

The measured values of s_0 for these two databases turned out to be 0.01 and 0.005, respectively.

6.2 Support and Distortion Settings

We evaluated the mining accuracy of MASK on the above datasets for a variety of sup_{min} and p values. We present here the results for two sup_{min} values, namely 0.25% and 0.5%. The 0.25% sup_{min} value represents, in a sense, the “worst-case” environment for our algorithm due to the presence of a large number of border-line itemsets.

The p values we consider are $p = 0.9$ and $p = 0.7$ (recall that these are equivalent to $p = 0.1$ and $p = 0.3$, respectively, with regard to privacy). For these settings, the associated privacy values for 1’s, as computed from Equation 1, are 85% and 96%, respectively, for the synthetic database, and 89% and 97%, respectively, for the real database.

6.3 Error Metrics

We evaluate two kinds of mining errors, Support Error and Identity Error, in our experiments:

Support Error (ρ) :

This metric reflects the (percentage) average relative error in the reconstructed support values for those itemsets that are correctly identified to be frequent. Denoting the reconstructed support by rec_sup and the actual support by act_sup , the support error is computed over all frequent itemsets as

$$\rho = \frac{1}{|f|} \sum_f \frac{|rec_sup_f - act_sup_f|}{act_sup_f} * 100$$

We compute this metric individually for each level of itemsets, that is, for 1-itemsets, 2-itemsets, etc.

Identity Error (σ) :

This metric reflects the percentage error in identifying frequent itemsets and has two components: σ^+ , indicating the percentage of false positives, and σ^- indicating the percentage of false negatives. Denoting the reconstructed set of frequent itemsets with R and the correct set of frequent itemsets with F , these metrics are computed as:

$$\sigma^+ = \frac{|R-F|}{|F|} * 100 \quad \sigma^- = \frac{|F-R|}{|F|} * 100$$

7 Experimental Results

We now present the results of our experiments conducted under the framework described above. The results for the synthetic database are presented first, followed by those for the real database.

7.1 Synthetic Database

Experiment 1: $p=0.9$, $sup_{min}=0.25\%$, $r=0\%$

Our first experiment was conducted on the synthetic database with a distortion parameter of $p=0.9$, $sup_{min}=0.25\%$, and no relaxation. As mentioned earlier, this experiment represents, in a sense, the worst case scenario for MASK due to the extremely low sup_{min} value.

The results for this experiment are shown in Table 2 – in this table, the level indicates the length of the frequent itemset, $|F|$ indicates the number of frequent itemsets at this level, and the other three columns are the error metrics defined in the previous section.

The results indicate that firstly the support error (ρ) is reasonably small, less than 5% at all levels. Secondly, the negative identity error is also small, not exceeding 6% at the maximum. Finally, the positive identity error is also in the same range. Note that the maximum errors occur for the 2-itemsets, which is as per expectations since the number of itemsets is the maximum at this level.

Level	$ F $	ρ	σ^-	σ^+
1	689	3.31	1.16	1.16
2	2648	3.58	4.49	5.14
3	1990	1.71	4.57	2.16
4	1418	1.28	3.67	0.22
5	730	1.27	5.89	0
6	212	1.36	4.25	5.19
7	35	1.40	0	0
8	3	0.99	0	0

Table 2: $p=0.9, sup_{min}=0.25\%, r=0\%$, Synthetic

Experiment 2: $p=0.9$, $sup_{min}=0.25\%$, $r=10\%$

Our second experiment evaluated the effect of marginally relaxing the sup_{min} value by 10% – that is, using a sup_{min} of 0.225% instead of 0.25%, keeping the rest of the parameters the same as that of Experiment 1. The ρ and σ results for this experiment are shown in Table 3.

Level	$ F $	ρ	σ^-	σ^+
1	689	3.37	0.73	3.19
2	2648	3.73	0.19	19.68
3	1990	1.76	0	28.09
4	1418	1.29	0	25.81
5	730	1.32	0	16.44
6	212	1.37	0	25.47
7	35	1.40	0	51.43
8	3	0.99	0	66.67

Table 3: $p=0.9, sup_{min}=0.25\%, r=10\%$, Synthetic

We see here that while the support error shows little change as compared to the previous experiment, the negative identity error goes down very significantly, becoming less than 1%, achieving the desired goal. The price to pay for this, however, is the substantial increase in the positive identity error. The marked change in the values of the identity errors also highlights the significant presence of “border-line” itemsets in the database.

Experiment 3: $p=0.9$, $sup_{min}=0.5\%$, $r=0\%$

We now move on to repeating Experiment 1 for an increased sup_{min} value of 0.5, corresponding to a “nicer” environment for MASK. The results of this experiment are shown in Table 4.

The results here show that the errors generally reduce as compared to Experiment 1 due to the sparser distribution of frequent itemsets. Further, 2-itemsets continue to be associated with the maximum error.

Experiment 4: $p=0.9$, $sup_{min}=0.5\%$, $r=10\%$

Our next experiment evaluated the effect of marginally relaxing the sup_{min} value by 10% – that is, using a

Level	$ F $	ρ	σ^-	σ^+
1	560	2.60	1.25	0.89
2	470	2.13	5.53	4.89
3	326	1.22	3.07	0.31
4	208	1.34	1.44	0.48
5	125	1.81	0	0
6	43	2.62	0	0
7	10	3.44	10	0
8	1	4.50	0	0

Table 4: $p=0.9, \text{sup}_{\min}=0.5\%, r=0\%$, Synthetic

sup_{\min} of 0.45% instead of 0.5% – keeping the rest of the parameters the same as that of Experiment 3. The ρ and σ results for this experiment are shown in Table 5.

Level	$ F $	ρ	σ^-	σ^+
1	560	2.66	0.18	4.29
2	470	2.21	0	44.89
3	326	1.26	0	42.64
4	208	1.35	0	51.44
5	125	1.81	0	22.4
6	43	2.62	0	18.60
7	10	3.47	0	10
8	1	4.50	0	0

Table 5: $p=0.9, \text{sup}_{\min}=0.5\%, r=10\%$, Synthetic

Similar to Experiment 2, the results here too indicate that a marginal relaxation can almost completely eliminate the false negative error component, ensuring that none of the true frequent itemsets are missed. At the same time, the false positive error goes up considerably since trying to “catch all the good fish” inevitably also attracts unwanted material.

Experiment 5: $p=0.7, \text{sup}_{\min}=0.25\%, r=10\%$

The previous experiments were all run at a distortion probability of $p = 0.9$ corresponding to a privacy factor of 85%. We now consider the possibility of improving the privacy measure to 96% by changing to $p = 0.7$ and evaluate the impact of this change on the accuracy. The results for this experiment are shown in Table 6. We see from these results that there is a dramatic increase in all the error metrics. For example, the support error goes up to about 25% on average, while the identity errors are huge. In fact, the errors go up to the extent that the results produced by the mining process are essentially meaningless. The implication is that privacy and accuracy represent an extremely sensitive tradeoff and that there is only a small parameter region wherein we can hope to obtain reasonable values for both metrics.

Level	$ F $	ρ	σ^-	σ^+
1	689	10.16	2.61	7.84
2	2648	25.23	19.52	630.93
3	1990	26.93	42.86	172.71
4	1418	29.14	65.94	0.35
5	730	28.47	79.32	0
6	212	36.25	84.91	0
7	35	51.37	85.71	0
8	3	–	100	0

Table 6: $p=0.7, \text{sup}_{\min}=0.25\%, r=10\%$, Synthetic

7.2 Real Database

We now move on to experiments conducted on the real database which, as mentioned earlier, contains information about the click-stream logs of an online leg-wear manufacturer. For ease of comparison, these experiments modeled exactly the same environments as those evaluated for the synthetic database – that is, Experiments 6 through 10 presented below correspond one-to-one with Experiments 1 through 5.

Experiment 6: $p=0.9, \text{sup}_{\min}=0.25\%, r=0\%$

Level	$ F $	ρ	σ^-	σ^+
1	249	5.89	4.02	2.81
2	239	3.87	6.69	7.11
3	73	2.60	10.96	9.59
4	4	1.41	0	25.0

Table 7: $p=0.9, \text{sup}_{\min}=0.25\%, r=0\%$, Real

The results of this experiment are shown in Table 7. We observe here that the errors, while still acceptably low, are somewhat higher than the corresponding numbers for the synthetic database. But, in fact, the absolute number of errors is fewer and it is due to the much smaller number of frequent itemsets that the effects of these errors are magnified. For example, the number of 2-itemsets in the real database is an order of magnitude smaller than that in the synthetic database.

Experiment 7: $p=0.9, \text{sup}_{\min}=0.25\%, r=10\%$

Level	$ F $	ρ	σ^-	σ^+
1	249	6.12	1.2	0.40
2	239	4.04	1.26	23.43
3	73	2.93	0	45.21
4	4	1.41	0	75

Table 8: $p=0.9, \text{sup}_{\min}=0.25\%, r=10\%$, Real

The results of this experiment are shown in Table 8. We observe that these results are similar to those of Experiment 2 – the basic errors are low and relaxation

significantly reduces the negative identity error at an attendant increase in the positive identity error.

Experiment 8: $p=0.9$, $sup_{min}=0.5\%$, $r=0\%$

Level	$ F $	ρ	σ^-	σ^+
1	150	4.23	0.67	4.67
2	45	2.42	2.22	4.44
3	6	1.07	0	16.66

Table 9: $p=0.9, sup_{min}=0.5\%, r=0\%, Real$

The results of this experiment are shown in Table 9. Here, we see that the stricter support threshold results in a very small set of frequent itemsets and that the errors are acceptably low.

Experiment 9: $p=0.9$, $sup_{min}=0.5\%$, $r=10\%$

Level	$ F $	ρ	σ^-	σ^+
1	150	4.27	0	8
2	45	2.56	0	37.77
3	6	1.07	0	66.66

Table 10: $p=0.9, sup_{min}=0.5\%, r=10\%, Real$

The results of this experiment are shown in Table 10. We see that the relaxation completely removes all false negatives, with the expected attendant increase in the false positives.

Experiment 10: $p=0.7$, $sup_{min}=0.25\%$, $r=10\%$

Level	$ F $	ρ	σ^-	σ^+
1	249	18.96	7.23	15.66
2	239	33.59	20.08	1907.53
3	73	32.87	30.14	2308.22
4	4	7.55	50	400

Table 11: $p=0.7, sup_{min}=0.25\%, r=10\%, Real$

The results of this experiment are shown in Table 11. These results are similar to those seen in Experiment 5 – the reduction in distortion probability results in a disproportionate increase in the errors for both the support and the identity metrics.

7.3 Summary

Overall, our experiments indicate that by a careful choice of distortion probability, it is possible to simultaneously achieve satisfactory privacy and accuracy. In particular, they show that there is a small “window of opportunity” around the $p = 0.9$ value where these dual goals can be met. Moving away from this window

towards lower values of p , however, results in skyrocketing errors, while increasing the value of p will result in significant loss of privacy.

We have conducted several other experiments with different market-basket type databases and the results are consistent with those presented here. One other issue is the running time of our mining algorithm. As mentioned earlier, mining the distorted database is intrinsically significantly more expensive than mining the real database. Currently, we have not yet implemented all the optimizations described in Section 5, and we have therefore refrained from presenting the running times since these numbers will change when all the optimizations are in place. The current situation is that we are able to mine the real database at 0.25% support in about 30 minutes on a low-end Pentium machine.

8 Reconstruction Error Bounds

As shown by our experiments of the previous section, mining the distorted database does result in errors in the reconstructed support. We now provide a loose probabilistic bound on these errors, focusing specifically on 1-itemsets since, as mentioned earlier, their errors percolate through the entire mining process.

Consider a single column in the true matrix. Suppose, it has n 1’s and $dbsize - n$ 0’s. We expect that the n 1’s will distort to np 1’s and $n(1 - p)$ 0’s when distorted with parameter p . Similarly, we expect the 0’s to go to $(dbsize - n)p$ 0’s and $(dbsize - n)(1 - p)$ 1’s. However, note that this is assuming that “When we generate a random number, which is distributed as $bernoulli(p)$, then the number of 1’s, denoted by P , in n trials is actually np ”. But, in reality, this will not be so. Actually, P is distributed as $binomial(n, p)$:

$$P = binomial(n, p) \\ P_r\{P = r, 0 \leq r \leq n\} = {}^nC_r p^r (1 - p)^{n-r}$$

As the value of n increases, the sample space of possible outcomes expands. And the probability that $P = np$ decreases. But, we are interested in an error bound around the value np . So, we define a function $P^E(n, p, \epsilon)$ as :

$$P^E(n, p, \epsilon) = P_r\{|Number\ of\ 1's - np| < \epsilon\}$$

That is, $P^E(n, p, \epsilon)$ measures the probability that when the above experiment is conducted, the number of 1’s is between $np - \epsilon$ and $np + \epsilon$. Clearly,

$$P^E(n, p, \epsilon) = \sum_{r=np-\epsilon}^{np+\epsilon} {}^nC_r p^r (1 - p)^{n-r}$$

Now, let the true column have n 1’s and m 0’s and the distorted column have n' 1’s and m' 0’s. Then, given the distorted column, MASK reconstructs the support to

$$\begin{aligned}\bar{n} &= \frac{pn'}{2p-1} - \frac{(1-p)m'}{2p-1} \\ \Rightarrow \bar{n} &= \frac{n'}{2p-1} - \frac{(1-p)dbsize}{2p-1}\end{aligned}\quad (6)$$

We now find the possible error in this approximation by using the function P^E . First, with probability $P^E(m, p, \epsilon_1)$, the number of 1's generated from the m 0's in the initial column is between $m(1-p) - \epsilon_1$ and $m(1-p) + \epsilon_1$. Next, with probability $P^E(n, p, \epsilon_2)$, the number of 1's generated from the n 1's in the initial column is between $np - \epsilon_2$ and $np + \epsilon_2$. Finally, with probability $P^E(m, p, \epsilon_1) \times P^E(n, p, \epsilon_2)$ the number of 1's in the distorted database (n') is between $m(1-p) + np - (\epsilon_1 + \epsilon_2)$ and $m(1-p) + np + (\epsilon_1 + \epsilon_2)$.

Note that the approximation in Equation 6 is based on the expectation that n' is equal to $m(1-p) + np$. Hence, with probability $P^E(m, p, \epsilon_1) \times P^E(n, p, \epsilon_2)$, we can assert that the error in the value of n' is

$$\Delta n' \leq \epsilon_1 + \epsilon_2$$

Now, from Equation 6, we know that the error in reconstruction is related to the error in the value of n' as follows:

$$\Delta \bar{n} = \frac{\Delta n'}{2p-1}$$

Therefore, we conclude that

$$\Rightarrow \Delta \bar{n} \leq \frac{\epsilon_1 + \epsilon_2}{2p-1}$$

In general, let $\epsilon_1 = \epsilon_2 = \frac{2p-1}{2}\epsilon$. Then we assert “With probability $P^E(m, p, \frac{2p-1}{2}\epsilon) \times P^E(n, p, \frac{2p-1}{2}\epsilon)$, the error in reconstruction is less than ϵ ”.

Note that in the above derivation, we have considered the worst case when both the errors are in the same direction, that is, deviating n' from the value $np + m(1-p)$ in the same direction. This is the reason for adding the values of ϵ_1 and ϵ_2 . In practice, however, we could expect that there is a reasonable likelihood that the errors follow opposite directions and therefore partially or fully negate each other.

9 Related Work

Concurrently with our work, the issue of maintaining privacy in association rule mining has attracted considerable attention over the last year [14, 5, 6, 15, 19, 11, 8]. The problem addressed in [14, 5, 6, 15] is how to prevent *sensitive rules* from being inferred by the data miner – the proposed solutions involve either falsifying some of the entries in the true database or replacing them with NULL values. This work is complementary to ours since it addresses concerns about *output* privacy, whereas our focus is on the privacy of the *input* data. Also note that, by definition, these techniques

require a completely materialized true database as the starting point whereas our approach can operate during the data collection process itself.

Maintaining input data privacy is considered in [19, 11] in the context of databases that are *distributed* across a number of sites with each site only willing to share data mining results, but not the source data. While [19] considers data that is vertically partitioned (i.e., each site hosts a disjoint subset of the matrix columns), the complementary situation where the data is horizontally partitioned (i.e., each site hosts a disjoint subset of the matrix rows) is addressed in [11]. The solution technique in [19] requires generating and computing a large set of independent linear equations – in fact, the number of equations and the number of terms in each equation is proportional to the *cardinality* of the database. It may therefore prove to be expensive for market-basket databases which typically contain millions of customer transactions. In [11], on the other hand, the problem is modeled as a secure multi-party computation [10] and an algorithm that minimizes the information shared without incurring much overhead on the mining process is presented. While these techniques are meaningful only in the context of distributed databases, our work is applicable to both centralized and distributed databases. Further, they assume a pre-existing true database at each site, whereas, as mentioned earlier, our approach can provide privacy directly at the user machine. Finally, a set of randomization operators for maintaining data privacy are presented and analyzed in [8].

Privacy-preserving mining in the context of *classification rules* has been investigated recently in [4, 1, 12]. Cryptographic protocols to ensure complete privacy in developing decision tree classifiers across distributed databases were presented in [12]. An alternative value distortion approach wherein a random value is added to each original value was taken in [4] and the privacy attained was quantified by the “fuzziness” provided by the system, that is, for a given level of confidence, the size of the interval that is expected to hold the original true value. Since we assume boolean data values, such an interval-based approach is not applicable in our context.

It was shown in [1] that the privacy estimates of [4] had to be lowered when the additional knowledge that the miner obtains from the reconstructed aggregate distribution was included in the problem formulation. The decrease is primarily due to the fact that values of the distorted data may lie *outside* the domain of the original data, thereby narrowing the interval associated with the true value. This problem cannot occur in our scenario, since both the original data and the distorted data have exactly the same domain, namely, 0 and 1. However, it is still possible, as explained very recently in [8], that the association rules forming the mining output may be used to re-interrogate

the distorted database and thereby reduce the privacy measure.

10 Conclusions

We have investigated the problem of supporting the conflicting goals of privacy and accuracy while mining association rules on large databases. Specifically, we presented a privacy metric and an analytical formula for evaluating the privacy of our MASK scheme, which is based on probabilistic distortion of user data, according to this metric. The formula showed that privacy is a function of the sparseness of the true matrix, as well as of the relative weight given to the privacy of 1's as compared to 0's.

A mining process for generating frequent itemsets from the distorted database was also presented, along with a set of optimizations to address the fact that mining the distorted database is significantly more expensive than mining the true database. The optimizations significantly reduced both the number of counters and the amount of counting that needs to be used in the mining process.

Our experimental results on synthetic and real databases showed that a distortion probability of $p = 0.9$ (equivalently, $p = 0.1$) is ideally suited to provide both privacy and good mining results for the sparse market-basket type of databases that we have considered in this study. Specifically, a privacy of over 80% and an error of less than 10% were simultaneously achieved with this setting.

In our future work, we plan to investigate the extension of our results to generalized [17] and quantitative [18] association rules. We also plan to refine our privacy estimation formulas to include the effects of using the mining output to re-interrogate the distorted database.

Acknowledgements

S. J. Rizvi was supported by a Summer Research Fellowship from the Centre for Theoretical Studies, Indian Institute of Science. J. R. Haritsa was supported by a research grant from the Dept. of Bio-technology, Govt. of India. We thank S. Chakrabarti of IIT-Bombay for his inputs during the early part of this work.

References

- [1] D. Agrawal and C. Aggarwal, "On the Design and Quantification of Privacy Preserving Data Mining Algorithms", *Proc. of 20th ACM Symp. on Principles of Database Systems (PODS)*, May 2001.
- [2] R. Agrawal, T. Imielinski and A. Swami, "Mining association rules between sets of items in large databases", *Proc. of ACM SIGMOD Intl. Conference on Management of Data (SIGMOD)*, May 1993.
- [3] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", *Proc. of 20th Intl. Conf. on Very Large Data Bases (VLDB)*, September 1994.
- [4] R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining", *Proc. of ACM SIGMOD Intl. Conf. on Management of Data*, May 2000.
- [5] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim and V. Verykios, "Disclosure Limitation of Sensitive Rules", *Proc. of IEEE Knowledge and Data Engineering Exchange Workshop (KDEX)*, November 1999.
- [6] E. Dasseni, V. Verykios, A. Elmagarmid and E. Bertino, "Hiding Association Rules by Using Confidence and Support", *Proc. of 4th Intl. Information Hiding Workshop (IHW)*, April 2001.
- [7] D. Denning, *Cryptography and Data Security*, Addison-Wesley, 1982.
- [8] A. Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, "Privacy Preserving Mining of Association Rules", *Proc. of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD)*, July 2002.
- [9] C. Hidber, "Online association rule mining", *Proc. of ACM SIGMOD Intl. Conf. on Management of Data*, June 1999.
- [10] O. Goldreich, "Secure Multi-party Computation", www.wisdom.weizmann.ac.il/~oded/pp.html, 1998.
- [11] M. Kantarcioglu and C. Clifton, "Privacy-preserving Distributed Mining of Association Rules on Horizontally Partitioned Data", *Proc. of ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD)*, June 2002.
- [12] Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining", *Advances in Cryptology - CRYPTO 2000*, Springer-Verlag LNCS 1880, 2000.
- [13] A. Savasere, E. Omiecinski and S. Navathe, "An efficient algorithm for mining association rules in large databases", *Proc. of 21st Intl. Conf. on Very Large Databases (VLDB)*, September 1995.
- [14] Y. Saygin, V. Verykios and C. Clifton, "Using Unknowns to Prevent Discovery of Association Rules", *ACM SIGMOD Record*, vol. 30, no. 4, 2001.
- [15] Y. Saygin, V. Verykios and A. Elmagarmid, "Privacy Preserving Association Rule Mining", *Proc. of 12th Intl. Workshop on Research Issues in Data Engineering (RIDE)*, February 2002.
- [16] P. Shenoy, J. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa and D. Shah, "Turbo-charging Vertical Mining of Large Databases", *Proc. of ACM SIGMOD Intl. Conf. on Management of Data*, May 2000.
- [17] R. Srikant and R. Agrawal, "Mining Generalized Association Rules", *Proc. of 21st Intl. Conf. on Very Large Databases (VLDB)*, September 1995.
- [18] R. Srikant and R. Agrawal, "Mining Quantitative Association Rules in Large Relational Tables", *Proc. of ACM SIGMOD Intl. Conference on Management of Data*, June 1996.
- [19] J. Vaidya and C. Clifton, "Privacy Preserving Association Rule Mining in Vertically Partitioned Data", *Proc. of 8th ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining (KDD)*, July 2002.
- [20] Z. Zheng, R. Kohavi and L. Mason, "Real World Performance of Association Rule Algorithms", *Proc. of 7th ACM SIGKDD Intl. Conference on Knowledge Discovery and Data Mining (KDD)*, August 2001.

Privacy Preserving Mining of Association Rules

Alexandre Evfimievski*

Ramakrishnan Srikant

Rakesh Agrawal

Johannes Gehrke*

IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120, USA

ABSTRACT

We present a framework for mining association rules from transactions consisting of categorical items where the data has been randomized to preserve privacy of individual transactions. While it is feasible to recover association rules and preserve privacy using a straightforward “uniform” randomization, the discovered rules can unfortunately be exploited to find privacy breaches. We analyze the nature of privacy breaches and propose a class of randomization operators that are much more effective than uniform randomization in limiting the breaches. We derive formulae for an unbiased support estimator and its variance, which allow us to recover itemset supports from randomized datasets, and show how to incorporate these formulae into mining algorithms. Finally, we present experimental results that validate the algorithm by applying it on real datasets.

1. INTRODUCTION

The explosive progress in networking, storage, and processor technologies is resulting in an unprecedented amount of digitization of information. It is estimated that the amount of information in the world is doubling every 20 months [20]. In concert with this dramatic and escalating increase in digital data, concerns about privacy of personal information have emerged globally [15] [17] [20] [24]. Privacy issues are further exacerbated now that the internet makes it easy for the new data to be automatically collected and added to databases [10] [13] [14] [27] [28] [29]. The concerns over massive collection of data are naturally extending to analytic tools applied to data. Data mining, with its promise to efficiently discover valuable, non-obvious information from large databases, is particularly vulnerable to misuse [11] [16] [20] [23].

An interesting new direction for data mining research is the development of techniques that incorporate privacy concerns [3]. The following question was raised in [7]: since the

*Department of Computer Science
Cornell University, Ithaca, NY 14853, USA

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGKDD 02 Edmonton, Alberta, Canada

Copyright 2002 ACM 1-58113-567-X/02/0007 ...\$5.00.

primary task in data mining is the development of models about aggregated data, can we develop accurate models without access to precise information in individual data records? Specifically, they studied the technical feasibility of building accurate classification models using training data in which the sensitive numeric values in a user’s record have been randomized so that the true values cannot be estimated with sufficient precision. Randomization is done using the statistical method of value distortion [12] that returns a value $x_i + r$ instead of x_i where r is a random value drawn from some distribution. They proposed a Bayesian procedure for correcting perturbed distributions and presented three algorithms for building accurate decision trees [9] [21] that rely on reconstructed distributions.¹ In [2], the authors derived an Expectation Maximization (EM) algorithm for reconstructing distributions and proved that the EM algorithm converged to the maximum likelihood estimate of the original distribution based on the perturbed data. They also pointed out that the EM algorithm was in fact identical to the Bayesian reconstruction procedure in [7], except for an approximation (partitioning values into intervals) that was made by the latter.

1.1 Contributions of this Paper

We continue the investigation of the use of randomization in developing privacy-preserving data mining techniques, and extend this line of inquiry along two dimensions:

- categorical data instead of numerical data, and
- association rule mining [4] instead of classification.

We will focus on the task of finding frequent itemsets in association rule mining, which we briefly review next.

Definition 1. Suppose we have a set \mathcal{I} of n items: $\mathcal{I} = \{a_1, a_2, \dots, a_n\}$. Let T be a sequence of N transactions $T = (t_1, t_2, \dots, t_N)$ where each transaction t_i is a subset of \mathcal{I} . Given an itemset $A \subset \mathcal{I}$, its *support* $\text{supp}^T(A)$ is defined as

$$\text{supp}^T(A) := \frac{\#\{t \in T \mid A \subseteq t\}}{N}. \quad (1)$$

An itemset $A \subset \mathcal{I}$ is called *frequent* in T if $\text{supp}^T(A) \geq \tau$, where τ is a user-defined parameter.

We consider the following setting. Suppose we have a server and many clients. Each client has a set of items (e.g.,

¹Once we have reconstructed distributions, it is straightforward to build classifiers that assume independence between attributes, such as Naive Bayes [19].

books or web pages or TV programs). The clients want the server to gather statistical information about associations among items, perhaps in order to provide recommendations to the clients. However, the clients do not want the server to know with certainty who has got which items. When a client sends its set of items to the server, it modifies the set according to some specific randomization policy. The server then gathers statistical information from the modified sets of items (transactions) and recovers from it the actual associations.

The following are the important results contained in this paper:

- In Section 2, we show that a straightforward uniform randomization leads to privacy breaches.
- We formally model and define privacy breaches in Section 3.
- We present a class of randomization operators in Section 4 that can be tuned for different tradeoffs between discoverability and privacy breaches. We derive formulae for the effect of randomization on support, and show how to recover the original support of an association from the randomized data.
- We present experimental results on two real datasets in Section 5, as well as graphs showing the relationship between discoverability, privacy, and data characteristics.

1.2 Related Work

There has been extensive research in the area of statistical databases motivated by the desire to provide statistical information (sum, count, average, maximum, minimum, p th percentile, etc.) without compromising sensitive information about individuals (see surveys in [1] [22].) The proposed techniques can be broadly classified into query restriction and data perturbation. The query restriction family includes restricting the size of query result, controlling the overlap amongst successive queries, keeping audit trail of all answered queries and constantly checking for possible compromise, suppression of data cells of small size, and clustering entities into mutually exclusive atomic populations. The perturbation family includes swapping values between records, replacing the original database by a sample from the same distribution, adding noise to the values in the database, adding noise to the results of a query, and sampling the result of a query. There are negative results showing that the proposed techniques cannot satisfy the conflicting objectives of providing high quality statistics and at the same time prevent exact or partial disclosure of individual information [1].

The most relevant work from the statistical database literature is the work by Warner [26], where he developed the “randomized response” method for survey results. The method deals with a single boolean attribute (e.g., drug addiction). The value of the attribute is retained with probability p and flipped with probability $1 \leftrightarrow p$. Warner then derived equations for estimating the true value of queries such as COUNT (Age = 42 & Drug Addiction = Yes). The approach we present in Section 2 can be viewed as a generalization of Warner’s idea.

Another related work is [25], where they consider the problem of mining association rules over data that is vertically partitioned across two sources, i.e, for each transaction, some of the items are in one source, and the rest in the

other source. They use multi-party computation techniques for scalar products to be able to compute the support of an itemset (when the two subsets that together form the itemset are in different sources), without either source revealing exactly which transactions support a subset of the itemset. In contrast, we focus on preserving privacy when the data is horizontally partitioned, i.e., we want to preserve privacy for individual transactions, rather than between two data sources that each have a vertical slice.

Related, but not directly relevant to our current work, is the problem of inducing decision trees over horizontally partitioned training data originating from sources who do not trust each other. In [16], each source first builds a local decision tree over its true data, and then swaps values amongst records in a leaf node of the tree to generate randomized training data. Another approach, presented in [18], does not use randomization, but makes use of cryptographic oblivious functions during tree construction to preserve privacy of two data sources.

2. UNIFORM RANDOMIZATION

A straightforward approach for randomizing transactions would be to generalize Warner’s “randomized response” method, described in Section 1.2. Before sending a transaction to the server, the client takes each item and with probability p replaces it by a new item not originally present in this transaction. Let us call this process *uniform* randomization.

Estimating true (nonrandomized) support of an itemset is nontrivial even for uniform randomization. Randomized support of, say, a 3-itemset depends not only on its true support, but also on the supports of its subsets. Indeed, it is much more likely that only one or two of the items are inserted by chance than all three. So, almost all “false” occurrences of the itemset are due to (and depend on) high subset supports. This requires estimating the supports of all subsets simultaneously. (The algorithm is similar to the algorithm presented in Section 4 for select-a-size randomization, and the formulae from Statements 1, 3 and 4 apply here as well.) For large values of p , most of the items in most randomized transactions will be “false”, so we seem to have obtained a reasonable privacy protection. Also, if there are enough clients and transactions, then frequent itemsets will still be “visible”, though less frequent than originally. For instance, after uniform randomization with $p = 80\%$, an itemset of 3 items that originally occurred in 1% transactions will occur in about $1\% \cdot (0.2)^3 = 0.008\%$ transactions, which is about 80 transactions per each million. The opposite effect of “false” itemsets becoming more frequent is comparatively negligible if there are many possible items: for 10,000 items, the probability that, say, 10 randomly inserted items contain a given 3-itemset is less than $10^{-7}\%$.

Unfortunately, this randomization has a problem. If we know that our 3-itemset escapes randomization in 80 per million transactions, and that it is unlikely to occur even once *because of* randomization, then every time we see it in a randomized transaction we know with near certainty of its presence in the nonrandomized transaction. With even more certainty we will know that at least one item from this itemset is “true”: as we have mentioned, a chance insertion of only one or two of the items is much more likely than of all three. In this case we can say that a *privacy breach* has occurred. Although privacy is preserved on average, personal information leaks through uniform randomization

for some fraction of transactions, despite the high value of p .

The rest of the paper is devoted to defining a framework for studying privacy breaches and developing techniques for finding frequent itemsets while avoiding breaches.

3. PRIVACY BREACHES

Definition 2. Let $(\Omega, \mathcal{F}, \mathbf{P})$ be a probability space of elementary events over some set Ω and σ -algebra \mathcal{F} . A *randomization operator* is a measurable function

$$R : \Omega \times \{\text{all possible } T\} \rightarrow \{\text{all possible } T\}$$

that randomly transforms a sequence of N transactions into a (usually) different sequence of N transactions. Given a sequence of N transactions T , we shall write $T' = R(T)$, where T is constant and $R(T)$ is a random variable.

Definition 3. Suppose that a nonrandomized sequence T is drawn from some known distribution, and $t_i \in T$ is the i -th transaction in T . A *general privacy breach of level ρ* with respect to a property $P(t_i)$ occurs if

$$\exists T' : \mathbf{P}[P(t_i) \mid R(T) = T'] \geq \rho.$$

We say that a property $Q(T')$ *causes a privacy breach of level ρ* with respect to $P(t_i)$ if

$$\mathbf{P}[P(t_i) \mid Q(R(T))] \geq \rho.$$

When we define privacy breaches, we think of the prior distribution of transactions as known, so that it makes sense to speak about a posterior probability of a property $P(t_i)$ versus prior. In practice, however, we do not know the prior distribution. In fact, there is no prior distribution; the transactions are not randomly generated. However, modeling transactions as being randomly generated from a prior distribution allows us to cleanly define privacy breaches.

Consider a situation when, for some transaction $t_i \in T$, an itemset $A \subseteq \mathcal{I}$ and an item $a \in A$, the property “ $A \subseteq t'_i \in T'$ ” causes a privacy breach w. r. t. the property “ $a \in t_i$.” In other words, the presence of A in a randomized transaction makes it likely that item a is present in the corresponding nonrandomized transaction.

Definition 4. We say that itemset A *causes a privacy breach of level ρ* if for some item $a \in A$ and some $i \in 1 \dots N$ we have $\mathbf{P}[a \in t_i \mid A \subseteq t'_i] \geq \rho$.

We will focus on controlling the class of privacy breaches given by Definition 4. Thus we ignore the effect of other information the server obtains from a randomized transaction, such as which items the randomized transaction does not contain, or the randomized transaction size. We also do not attempt to control breaches that occur because the server knows some other information about items and clients besides the transactions. For example, the server may know some geographical or demographic data about the clients. Finally, in Definition 4, we only considered positive breaches, i.e., we know with high probability that an item was present in the original transaction. In some scenarios, being confident that an item was *not* present in the original transaction may also be considered a privacy breach.

4. ALGORITHM

“Where does a wise man hide a leaf? In the forest. But what does he do if there is no forest?”
... “He grows a forest to hide it in.” – G.K. Chesterton, “The Sign of the Broken Sword”

The intuition of breach control is quite simple: in addition to replacing some of the items, we shall insert so many “false” items into a transaction that one is as likely to see a “false” itemset as a “true” one.

4.1 Randomization Operators

Definition 5. We call randomization R a *per-transaction randomization* if, for $T = (t_1, t_2, \dots, t_N)$, we can represent $R(T)$ as

$$R(t_1, t_2, \dots, t_N) = (R(1, t_1), R(2, t_2), \dots, R(N, t_N)),$$

where $R(i, t)$ are independent random variables whose distributions depend only on t (and not on i). We shall write $t'_i = R(i, t_i) = R(t_i)$.

Definition 6. A randomization operator R is called *item-invariant* if, for every transaction sequence T and for every permutation $\pi : \mathcal{I} \rightarrow \mathcal{I}$ of items, the distribution of $\pi^{-1}R(\pi T)$ is the same as of $R(T)$. Here πT means the application of π to all items in all transactions of T at once.

Definition 7. A *select-a-size* randomization operator has the following parameters, for each possible input transaction size m :

- Default probability of an item (also called *randomization level*) $\rho_m \in (0, 1)$;
- Transaction subset size selection probabilities $p_m[0], p_m[1], \dots, p_m[m]$, such that every $p_m[j] \geq 0$ and

$$p_m[0] + p_m[1] + \dots + p_m[m] = 1.$$

Given a sequence of transactions $T = (t_1, t_2, \dots, t_N)$, the operator takes each transaction t_i independently and proceeds as follows to obtain transaction t'_i ($m = |t_i|$).

1. The operator selects an integer j at random from the set $\{0, 1, \dots, m\}$ so that $\mathbf{P}[j \text{ is selected}] = p_m[j]$.
2. It selects j items from t_i , uniformly at random (without replacement). These items, and no other items of t_i , are placed into t'_i .
3. It considers each item $a \notin t_i$ in turn and tosses a coin with probability ρ_m of “heads” and $1 \Leftrightarrow \rho_m$ of “tails”. All those items for which the coin faces “heads” are added to t'_i .

Remark 1. Both uniform (Section 2) and select-a-size operators are per-transaction because they apply the same randomization algorithm to each transaction independently. They are also item-invariant since they do not use any item-specific information (if we rename or reorder the items, the outcome probabilities will not be affected).

Definition 8. A *cut-and-paste* randomization operator is a special case of a select-a-size operator (and which we shall actually test on datasets). For each possible input transaction size m , it has two parameters: $\rho_m \in (0, 1)$ (randomization level) and an integer $K_m > 0$ (the *cutoff*). The operator takes each input transaction t_i independently and proceeds as follows to obtain transaction t'_i (here $m = |t_i|$):

1. It chooses an integer j uniformly at random between 0 and K_m ; if $j > m$, it sets $j = m$.
2. The operator selects j items out of t_i uniformly at random (without replacement). These items are placed into t'_i .
3. Each other item (including the rest of t_i) is placed into t'_i with probability ρ_m , independently.

Remark 2. For any m , a cut-and-paste operator has only two parameters, ρ_m and K_m , to play with; moreover, K_m is an integer. Because it is easy to find optimal values for these parameters (Section 4.4), we chose to test this operator, leaving open the problem of optimizing the m parameters of the “unabridged” select-a-size. To see that cut-and-paste is a case of select-a-size, let us write down the formulae for the $p_m[j]$ ’s:

$$p_m[j] = \sum_{i=0}^{\min\{K, j\}} \binom{m \Leftrightarrow i}{j \Leftrightarrow i} \rho^{j-i} (1 \Leftrightarrow \rho)^{m-j} \cdot \begin{cases} 1 \Leftrightarrow m/(K+1) & \text{if } i = m \text{ and } i < K \\ 1/(K+1) & \text{otherwise} \end{cases}$$

Now let us give one example of a randomization operator that is not a per-transaction randomization, because it uses the knowledge of several transactions per each randomized transaction.

Example 1. The *mixing* randomization operator has one integer parameter $K \geq 2$ and one real-valued parameter $p \in (0, 1)$. Given a sequence of transactions $T = (t_1, t_2, \dots, t_N)$, the operator takes each transaction t_i independently and proceeds as follows to obtain transaction t'_i :

1. Other than t_i , pick $K \Leftrightarrow 1$ more transactions (with replacement) from T and union the K transactions as sets of items. Let t''_i be this union.
2. Consider each item $a \in t''_i$ in turn and toss a coin with probability p of “heads” and $1 \Leftrightarrow p$ of “tails”.
3. All those items for which the coin faces “tails” are removed from the transaction. The remaining items constitute the randomized transaction.

For the purpose of privacy-preserving data mining, it is natural to focus mostly on per-transaction randomizations, since they are the easiest and safest to implement. Indeed, a per-transaction randomization does not require the users (who submit randomized transactions to the server) to communicate with each other in any way, nor to exchange random bits. On the contrary, implementing mixing randomization, for example, requires to organize an exchange of nonrandomized transactions between users, which opens an opportunity for cheating or eavesdropping.

4.2 Effect of Randomization on Support

Let T be a sequence of transactions of length N , and let A be some subset of items (that is, $A \subseteq \mathcal{I}$). Suppose we randomize T and get $T' = R(T)$. The support $s' = \text{supp}^{T'}(A)$ of A for T' is a random variable that depends on the outcome of randomization. Here we are going to determine the distribution of s' , under the assumption of having a per-transaction and item-invariant randomization.

Definition 9. The fraction of the transactions in T that have intersection with A of size l among all transactions in T is called *partial support* of A for intersection size l :

$$\text{supp}_l^T(A) := \frac{\#\{t \in T \mid \#(A \cap t) = l\}}{N}. \quad (2)$$

It is easy to see that $\text{supp}^T(A) = \text{supp}_k^T(A)$ for $k = |A|$, and that

$$\sum_{l=0}^k \text{supp}_l^T(A) = 1$$

since those transactions in T that do not intersect A at all are covered in $\text{supp}_0^T(A)$.

Definition 10. Suppose that our randomization operator is both per-transaction and item-invariant. Consider a transaction t of size m and an itemset $A \subset \mathcal{I}$ of size k . After randomization, transaction t becomes t' . We define

$$p_k^m[l \rightarrow l'] = p[l \rightarrow l'] := \mathbf{P}[\#(t' \cap A) = l' \mid \#(t \cap A) = l]. \quad (3)$$

Here both l and l' must be integers in $\{0, 1, \dots, k\}$.

Remark 3. The value of $p_k^m[l \rightarrow l']$ is well-defined (does not depend on any other information about t and A , or other transactions in T and T' besides t and t'). Indeed, because we have a per-transaction randomization, the distribution of t' depends neither on other transactions in T besides t , nor on their randomized outcomes. If there were other t_1 and B with the same (m, k, l) , but a different probability (3) for the same l' , we could consider a permutation π of \mathcal{I} such that $\pi t = t_1$ and $\pi A = B$; the application of π or of π^{-1} would preserve intersection sizes l and l' . By item-invariance we have

$$\mathbf{P}[\#(t' \cap A) = l'] = \mathbf{P}[\#(\pi^{-1}R(\pi t) \cap A) = l'],$$

but by the choice of π we also have

$$\begin{aligned} \mathbf{P}[\#(\pi^{-1}R(\pi t) \cap A) = l'] &= \mathbf{P}[\#(\pi^{-1}R(t_1) \cap \pi^{-1}B) = l'] \\ &= \mathbf{P}[\#(t'_1 \cap B) = l'] \neq \mathbf{P}[\#(t' \cap A) = l'], \end{aligned}$$

a contradiction.

STATEMENT 1. Suppose that our randomization operator is both per-transaction and item-invariant. Suppose also that all the N transactions in T have the same size m . Then, for a given subset $A \subseteq \mathcal{I}$, $|A| = k$, the random vector

$$N \cdot (s'_0, s'_1, \dots, s'_k), \quad \text{where } s'_l := \text{supp}_l^{T'}(A) \quad (4)$$

is a sum of $k+1$ independent random vectors, each having a multinomial distribution. Its expected value is given by

$$\mathbf{E}(s'_0, s'_1, \dots, s'_k)^T = P \cdot (s_0, s_1, \dots, s_k)^T \quad (5)$$

where P is the $(k+1) \times (k+1)$ matrix with elements $P_{l',l} = p[l \rightarrow l']$, and the covariance matrix is given by

$$\text{Cov}(s'_0, s'_1, \dots, s'_k)^T = \frac{1}{N} \cdot \sum_{l=0}^k s_l D[l] \quad (6)$$

where each $D[l]$ is a $(k+1) \times (k+1)$ matrix with elements

$$D[l]_{i,j} = p[l \rightarrow i] \cdot \delta_{i=j} \Leftrightarrow p[l \rightarrow i] \cdot p[l \rightarrow j]. \quad (7)$$

Here s_l denotes $\text{supp}_l^T(A)$, and the T over vectors denotes the transpose operation; $\delta_{i=j}$ is one if $i = j$ and zero otherwise.

PROOF. See Appendix A.1. \square

Remark 4. In Statement 1 we have assumed that all transactions in T have the same size. If this is not so, we have to consider each transaction size separately and then use per-transaction independence.

STATEMENT 2. For a select- a -size randomization with randomization level ρ and size selection probabilities $\{p_m[j]\}$, we have:

$$p_k^m[l \rightarrow l'] = \sum_{j=0}^m p_m[j] \cdot \sum_{q=\max\{0, j+l-m, l+l'-k\}}^{\min\{j, l, l'\}} \frac{\binom{l}{q} \binom{m \Leftrightarrow l}{j \Leftrightarrow q}}{\binom{m}{j}} \cdot \left(\frac{k \Leftrightarrow l}{l' \Leftrightarrow q} \right) \rho^{l'-q} (1 \Leftrightarrow \rho)^{k-l-l'+q}. \quad (8)$$

PROOF. See Appendix A.2. \square

4.3 Support Recovery

Let us assume that all transactions in T have the same size m , and let us denote

$$\vec{s} := (s_0, s_1, \dots, s_k)^T, \quad \vec{s}' := (s'_0, s'_1, \dots, s'_k)^T;$$

then, according to (5), we have

$$\mathbf{E} \vec{s}' = P \cdot \vec{s}. \quad (9)$$

Denote $Q = P^{-1}$ (assume that it exists) and multiply both sides of (9) by Q :

$$\vec{s} = Q \cdot \mathbf{E} \vec{s}' = \mathbf{E} Q \cdot \vec{s}'.$$

We have thus obtained an unbiased estimator for the original partial supports given randomized partial supports:

$$\vec{s}_{\text{est}} := Q \cdot \vec{s}' \quad (10)$$

Using (6), we can compute the covariance matrix of \vec{s}_{est} :

$$\begin{aligned} \text{Cov} \vec{s}_{\text{est}} &= \text{Cov}(Q \cdot \vec{s}') = Q (\text{Cov} \vec{s}') Q^T = \\ &= \frac{1}{N} \cdot \sum_{l=0}^k s_l Q D[l] Q^T. \end{aligned} \quad (11)$$

If we want to estimate this covariance matrix by looking only at randomized data, we may use \vec{s}_{est} instead of \vec{s} in (11):

$$(\text{Cov} \vec{s}_{\text{est}})_{\text{est}} = \frac{1}{N} \cdot \sum_{l=0}^k (\vec{s}_{\text{est}})_l Q D[l] Q^T.$$

This estimator is also unbiased:

$$\mathbf{E}(\text{Cov} \vec{s}_{\text{est}})_{\text{est}} = \frac{1}{N} \cdot \sum_{l=0}^k (\mathbf{E} \vec{s}_{\text{est}})_l Q D[l] Q^T = \text{Cov} \vec{s}_{\text{est}}.$$

In practice, we want only the k -th coordinate of \vec{s} , that is, the support $s = \text{supp}^T(A)$ of our itemset A in T . We denote by \tilde{s} the k -th coordinate of \vec{s}_{est} , and use \tilde{s} to estimate s . Let us compute simple formulae for \tilde{s} , its variance and the unbiased estimator of its variance. Denote

$$q[l \leftarrow l'] := Q_{l,l'}.$$

STATEMENT 3.

$$\tilde{s} = \sum_{l'=0}^k s'_{l'} \cdot q[k \leftarrow l'];$$

$$\text{Var} \tilde{s} = \frac{1}{N} \sum_{l=0}^k s_l \left(\sum_{l'=0}^k p[l \rightarrow l'] q[k \leftarrow l']^2 \Leftrightarrow \delta_{l=k} \right);$$

$$(\text{Var} \tilde{s})_{\text{est}} = \frac{1}{N} \sum_{l'=0}^k s'_{l'} (q[k \leftarrow l']^2 \Leftrightarrow q[k \leftarrow l']).$$

PROOF. See Appendix A.3. \square

We conclude this subsection by giving a linear coordinate transformation in which the matrix P from Statement 1 becomes triangular. (We use this transformation for privacy breach analysis in Section 4.4.) The coordinates after the transformation have a combinatorial meaning, as given in the following definition.

Definition 11. Suppose we have a transaction sequence T and an itemset $A \subseteq \mathcal{I}$. Given an integer l between 0 and $k = |A|$, consider all subsets $C \subseteq A$ of size l . The sum of supports of all these subsets is called the *cumulative support* for A of order l and is denoted as follows:

$$\begin{aligned} \Sigma_l &= \Sigma_l(A, T) := \sum_{C \subseteq A, |C|=l} \text{supp}^T(C), \\ \vec{\Sigma} &:= (\Sigma_0, \Sigma_1, \dots, \Sigma_k)^T \end{aligned} \quad (12)$$

STATEMENT 4. The vector $\vec{\Sigma}$ of cumulative supports is a linear transformation of the vector \vec{s} of partial supports, namely,

$$\Sigma_l = \sum_{j=l}^k \binom{j}{l} s_j \quad \text{and} \quad s_l = \sum_{j=l}^k (\Leftrightarrow 1)^{j-l} \binom{j}{l} \Sigma_j; \quad (13)$$

in the $\vec{\Sigma}$ and $\vec{\Sigma}'$ space (instead of \vec{s} and \vec{s}') matrix P is lower triangular.

PROOF. See Appendix A.4. \square

4.4 Limiting Privacy Breaches

Here we determine how privacy depends on randomization. We shall use Definition 4 and assume a per-transaction and item-invariant randomization.

Consider some itemset $A \subseteq \mathcal{I}$ and some item $a \in A$; fix a transaction size m . We shall assume that m is known to the server, so that we do not have to combine probabilities

for different nonrandomized sizes. Assume also that a partial support $s_l = \text{supp}_l^T(A)$ approximates the corresponding prior probability $\mathbf{P}[\#(t \cap A) = l]$. Suppose we know the following prior probabilities:

$$\begin{aligned} s_l^+ &:= \mathbf{P}[\#(t \cap A) = l, a \in t], \\ s_l^- &:= \mathbf{P}[\#(t \cap A) = l, a \notin t]. \end{aligned}$$

Notice that $s_l = s_l^+ + s_l^-$ simply because

$$\#(t \cap A) = l \Leftrightarrow \begin{cases} a \in t \ \& \ \#(t \cap A) = l, \text{ or} \\ a \notin t \ \& \ \#(t \cap A) = l. \end{cases}$$

Let us use these priors and compute the posterior probability of $a \in t$ given $A \subseteq t'$:

$$\begin{aligned} \mathbf{P}[a \in t \mid A \subseteq t'] &= \frac{\mathbf{P}[a \in t, A \subseteq t']}{\mathbf{P}[A \subseteq t']} = \\ &= \frac{\sum_{l=1}^k \mathbf{P}[\#(t \cap A) = l, a \in t, A \subseteq t']}{\sum_{l=0}^k s_l \cdot p[l \rightarrow k]} \\ &= \frac{\sum_{l=1}^k \mathbf{P}[\#(t \cap A) = l, a \in t] \cdot p[l \rightarrow k]}{\sum_{l=0}^k s_l \cdot p[l \rightarrow k]} \\ &= \frac{\sum_{l=1}^k s_l^+ \cdot p[l \rightarrow k]}{\sum_{l=0}^k s_l \cdot p[l \rightarrow k]}. \end{aligned}$$

Thus, in order to prevent privacy breaches of level 50% as defined in Definition 4, we need to ensure that always

$$\sum_{l=1}^k s_l^+ \cdot p[l \rightarrow k] < 0.5 \cdot \sum_{l=0}^k s_l \cdot p[l \rightarrow k]. \quad (14)$$

The problem is that we have to randomize the data *before* we know any supports. Also, we may not have the luxury of setting “oversafe” randomization parameters because then we may not have enough data to perform a reasonably accurate support recovery. One way to achieve a compromise is to:

1. Estimate maximum possible support $s_{\max}(k, m)$ of a k -itemset in the transactions of given size m , for different k and m ;
2. Given the maximum supports, find values for s_l and s_l^+ that are most likely to cause a privacy breach;
3. Make randomization just strong enough to prevent such a privacy breach.

Since $s_0^+ = 0$, the most privacy-challenging situations occur when s_0 is small, that is, when our itemset A and its subsets are frequent.

In our experiments we consider a privacy-challenging k -itemset A such that, for every $l > 0$, all its subsets of size l have the maximum possible support $s_{\max}(l, m)$. The partial supports for such a test-itemset are computed from the cumulative supports Σ_l using Statement 4. By it and by (12), we have ($l > 0$)

$$s_l = \sum_{j=l}^k (\Leftrightarrow 1)^{j-l} \binom{j}{l} \Sigma_j, \quad \Sigma_j = \binom{k}{j} s_{\max}(j, m) \quad (15)$$

since there are $\binom{k}{j}$ j -subsets in A . The values of s_l^+ follow if we note that all l -subsets of A , with a and without, appear

equally frequently as $t \cap A$:

$$\begin{aligned} s_l^+ &:= \mathbf{P}[\#(t \cap A) = l, a \in t] = \\ &= \mathbf{P}[a \in t \mid \#(t \cap A) = l] \cdot s_l = l/k \cdot s_l. \end{aligned} \quad (16)$$

While one can construct cases that are even more privacy-challenging (for example, if $a \in A$ occurs in a transaction every time any nonempty subset of A does), we found the above model (15) and (16) to be sufficiently pessimistic on our datasets.

We can now use these formulae to obtain cut-and-paste randomization parameters ρ_m and K_m as follows. Given m , consider all cutoffs from $K_m = 3$ to some K_{\max} (usually this K_{\max} equals the maximum transaction size) and determine the smallest randomization levels $\rho_m(K_m)$ that satisfy (14). Then select (K_m, ρ_m) that gives the best discoverability (by computing the lowest discoverable supports, see Section 5.1).

4.5 Discovering Associations

We show how to discover itemsets with high true support given a set of randomized transactions. Although we use the Apriori algorithm [5] to make the ideas concrete, the modifications directly apply to any algorithm that uses Apriori candidate generation, i.e., to most current association discovery algorithms.² The key *lattice property* of supports used by *Apriori* is that, for any two itemsets $A \subseteq B$, the true support of A is equal to or larger than the true support of B . A simplified version of *Apriori*, given a (nonrandomized) transactions file and a minimum support s_{\min} , works as follows:

1. Let $k = 1$, let “candidate sets” be all single items. Repeat the following until no candidate sets are left:
 - (a) Read the data file and compute the supports of all candidate sets;
 - (b) Discard all candidate sets whose support is below s_{\min} ;
 - (c) Save the remaining candidate sets for output;
 - (d) Form all possible $(k + 1)$ -itemsets such that all their k -subsets are among the remaining candidates. Let these itemsets be the new candidate sets.
 - (e) Let $k = k + 1$.
2. Output all the saved itemsets.

It is (conceptually) straightforward to modify this algorithm so that now it reads the randomized dataset, computes partial supports of all candidate sets (for all nonrandomized transaction sizes) and recovers their predicted supports and sigmas using the formulae from Statement 3. However, for the predicted supports the lattice property is no longer true. It is quite likely that for an itemset that is slightly above minimum support and whose predicted support is also above minimum support, that one of its subsets will have predicted support below minimum support. So if we discard all candidates below minimum support for the purpose of candidate generation, we will miss many (perhaps even the majority)

²The main class of algorithms where this would not apply are those that find only maximal frequent itemsets, e.g., [8]. However, randomization precludes finding very long itemsets, so this is a moot point.

of the longer frequent itemsets. Hence, for candidate generation, we discard only those candidates whose predicted support is “significantly” smaller than s_{\min} , where significance is measured by means of predicted sigmas. Here is the modified version of *Apriori*:

1. Let $k = 1$, let “candidate sets” be all single-item sets. Repeat the following until k is too large for support recovery (or until no candidate sets are left):
 - (a) Read the randomized data file and compute the partial supports of all candidate sets, separately for each nonrandomized transaction size³;
 - (b) Recover the predicted supports and sigmas for the candidate sets;
 - (c) Discard every candidate set whose support is below its *candidate limit*;
 - (d) Save for output only those candidate sets whose predicted support is at least s_{\min} ;
 - (e) Form all possible $(k + 1)$ -itemsets such that all their k -subsets are among the remaining candidates. Let these itemsets be the new candidate sets.
 - (f) Let $k = k + 1$.
2. Output all the saved itemsets.

We tried $s_{\min} \leftrightarrow \sigma$ and $s_{\min} \leftrightarrow 2\sigma$ as the candidate limit, and found that the former does a little better than the latter. It prunes more itemsets and therefore makes the algorithm work faster, and, when it discards a subset of an itemset with high predicted support, it usually turns out that the true support of this itemset is not as high.

5. EXPERIMENTAL RESULTS

Before we come to the experiments with datasets, we first show in Section 5.1 how our ability to recover supports depends on the permitted breach level, as well as other data characteristics. We then describe the real-life datasets in Section 5.2, and present results on these datasets in Section 5.3.

5.1 Privacy, Discoverability and Dataset Characteristics

We define the *lowest discoverable support* as the support at which the predicted support of an itemset is four sigmas away from zero, i.e., we can clearly distinguish the support of this itemset from zero. In practice, we may achieve reasonably good results even if the minimum support level is slightly lower than four sigma (as was the case for 3-itemsets in the randomized *soccer*, see below). However, the lowest discoverable support is a nice way to illustrate the interaction between discoverability, privacy breach levels, and data characteristics.

Figure 1 shows how the lowest discoverable support changes with the privacy breach level. For higher privacy breach levels such as 95% (which could be considered a “plausible denial” breach level), we can discover 3-itemsets at very low supports. For more conservative privacy breach levels

³In our experiments, the nonrandomized transaction size is always known and included as a field into every randomized transaction

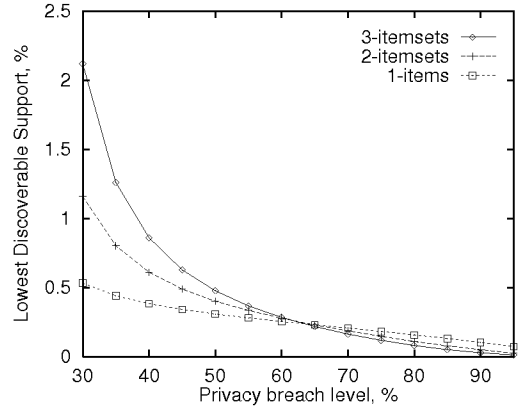


Figure 1: Lowest discoverable support for different breach levels. Transaction size is 5, five million transactions.

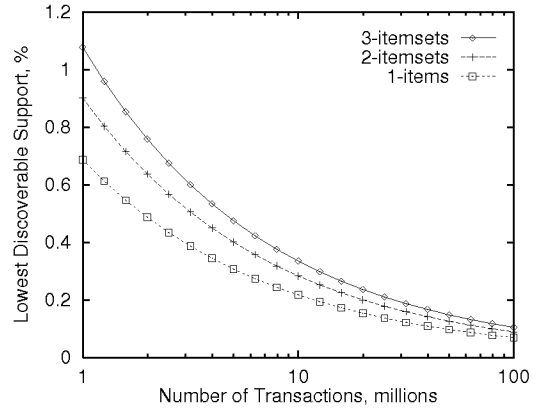


Figure 2: Lowest discoverable support versus number of transactions. Transaction size is 5, breach level is 50%.

such as 50%, the lowest discoverable support is significantly higher. It is interesting to note that at higher breach levels (i.e. weaker randomization) it gets harder to discover 1-itemset supports than 3-itemset supports. This happens because the variance of a 3-itemset predictor depends highly nonlinearly on the amount of false items added while randomizing. When we add fewer false items at higher breach levels, we generate so much fewer false 3-itemset positives than false 1-itemset positives that 3-itemsets get an advantage over single items.

Figure 2 shows that the lowest discoverable support is roughly inversely proportional to the square root of the number of transactions. Indeed, the lowest discoverable support is defined to be proportional to the standard deviation (square root of the variance) of this support’s prediction. If all the partial supports are fixed, the prediction’s variance is inversely proportional to the number N of transactions according to Statement 3. In our case, the partial supports depend on N (because the lowest discoverable support does), i.e. they are not fixed; however, this does not appear to affect the variance very significantly (but justifies the word “roughly”).

Finally, Figure 3 shows that transaction size has a sig-

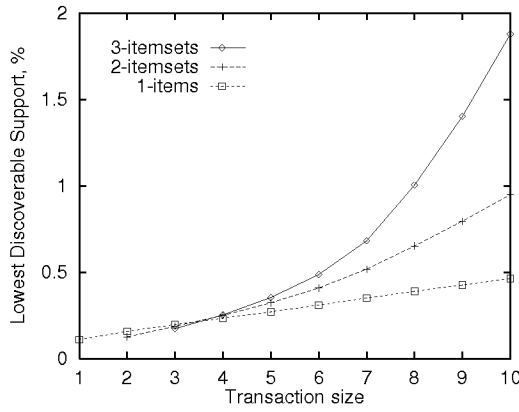


Figure 3: Lowest discoverable support for different transaction sizes. Five million transactions, breach level is 50%.

nificant influence on support discoverability. In fact, for transactions of size 10 and longer, it is typically not possible to make them both breach-safe and simultaneously get useful information for mining transactions. Intuitively, a long transaction contains too much personal information to hide, because it may contain long frequent itemsets whose appearance in the randomized transaction could result in a privacy breach. We have to insert a lot of false items and cut off many true ones to ensure that such a long itemset in the randomized transaction is about as likely to be a false positive as to be a true positive. Such a strong randomization causes an exceedingly high variance in the support predictor for 2- and especially 3-itemsets, since it drives down their probability to “tunnel” through while raising high the probability of a false positive. In both our datasets we discard long transactions. The question of how to safely randomize and mine long transactions is left open.

5.2 The Datasets

We experimented with two “real-life” datasets. The **soccer** dataset is generated from the clickstream log of the 1998 World Cup Web site, which is publicly available at <ftp://research.smp2.cc.vt.edu/pub/worldcup/>⁴. We scanned the log and produced a transaction file, where each transaction is a session of access to the site by a client. Each item in the transaction is a web request. Not all web requests were turned into items; to become an item, the request must satisfy the following:

1. Client’s request method is **GET**;
2. Request status is **OK**;
3. File type is **HTML**.

A session starts with a request that satisfies the above properties, and ends when the last click from this ClientID time-outs. The timeout is set as 30 minutes. All requests in a session have the same ClientID. The **soccer** transaction file was then processed further: we deleted from all transactions the items corresponding to the French and English front page frames, and then we deleted all empty transactions and all transactions of size above 10. The resulting **soccer** dataset

⁴M. Arlitt and T. Jin, “1998 World Cup Web Site Access Logs”, August 1998. Available at <http://www.acm.org/sigcomm/ITA/>

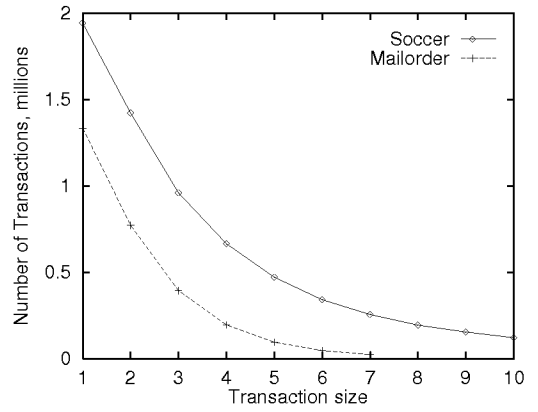


Figure 4: Number of transactions for each transaction size in the soccer and mailorder datasets.

consists of 6,525,879 transactions, distributed as shown in Fig. 4.

The **mailorder** dataset is the same as that used in [6]. The original dataset consisted of around 2.9 million transactions, 15,836 items, and around 2.62 items per transaction. Each transaction was the set of items purchased in a single mail order. However, very few itemsets had reasonably high supports. For instance, there were only two 2-itemsets with support $\geq 0.2\%$, only five 3-itemsets with support $\geq 0.05\%$. Hence we decided to substitute all items by their parents in the taxonomy, which had reduced the number of items from 15836 to 96. It seems that, in general, moving items up the taxonomy is a natural thing to do for preserving privacy without losing aggregate information. We also discarded all transactions of size ≥ 8 (which was less than 1% of all transactions) and finally obtained a dataset containing 2,859,314 transactions (Fig. 4).

5.3 The Results

We report the results for both datasets at a minimum support that is close to the lowest discoverable support, in order to show the resilience of our algorithm even at these very low support levels. We targeted a conservative breach level of 50%, so that, given a randomized transaction, for any item in the transaction it is at least as likely that someone did not buy that item (or access a web page) as that they did buy that item.

We used cut-and-paste randomization (see Definition 8) that has only two parameters, randomization level and cut-off, per each transaction size. We chose a cutoff of 7 for our experiments as a good compromise between privacy and discoverability. Given the values of maximum supports, we then used the methodology from Section 4.4 to find the lowest randomization level such that the breach probability (for each itemset size) is still below the desired breach level. The actual parameters (K_m is the cutoff, ρ_m is the randomization level for transaction size m) for **soccer** were:

m	1	2	3	4	5	6	7	8	9	10
K_m	7	7	7	7	7	7	7	7	7	7
$\rho_m\%$	4.7	16.8	21.4	32.2	35.3	42.9	46.1	42.0	40.9	39.5

and for **mailorder** were:

m	1	2	3	4	5	6	7
K_m	7	7	7	7	7	7	7
$\rho_m\%$	8.9	20.4	25.0	33.4	43.5	50.5	59.2

Table 1 shows what happens if we mine itemsets from both randomized and nonrandomized files and then compare the results. We can see that, even for a low minimum support of 0.2%, most of the itemsets are mined correctly from the randomized file. There are comparatively few false positives (itemsets wrongly included into the output) and even fewer false drops (itemsets wrongly omitted). The predicted sigma for 3-itemsets ranges in $0.066 \pm 0.07\%$ for *soccer* and in $0.047 \pm 0.048\%$ for *mailorder*; for 2- and 1-itemsets sigmas are even less.

One might be concerned about the true supports of the false positives. Since we know that there are *many* more low-supported itemsets than there are highly supported, we might wonder whether most of the false positives are outliers, that is, have true support near zero. We have indeed seen outliers; however, it turns out that most of the false positives are not so far off. The tables 2 and 3 show that usually the true supports of false positives, as well as the predicted supports of false drops, are closer to 0.2% than to zero. This good news demonstrates the promise of randomization as a practical privacy-preserving approach.

Privacy Analysis We evaluate privacy breaches, i.e., the conditional probabilities from Definition 4, as follows. We count the occurrences of an itemset in a randomized transaction and its sub-items in the corresponding nonrandomized transaction. For example, assume an itemset $\{a, b, c\}$ occurs 100 times in the randomized data among transactions of length 5. Out of these 100 occurrences, 60 of the corresponding original transactions had the item b . We then say that this itemset caused a 60% privacy breach for transactions of length 5, since for these 100 randomized transactions, we estimate with 60% confidence that the item b was present in the original transaction.

Out of all sub-items of an itemset, we choose the item that causes the worst privacy breach. Then, for each combination of transaction size and itemset size, we compute over all frequent⁵ itemsets the worst and the average value of this breach level. Finally, we pick the itemset size that gave the worst value for each of these two values.

Table 4 shows the results of the above analysis. To the left of the semicolon is the itemset size that was the worst. For instance, for all transactions of length 5 for *soccer*, the worst average breach was with 4-itemsets (43.9% breach), and the worst breach was with a 5-itemset (49.7% breach). We can see that, apart from fluctuations, the 50% level is observed everywhere except of a little “slip” for 9- and 10-item transactions of *soccer*. The “slip” resulted from our decision to use the corresponding maximal support information only for itemset sizes up to 7 (while computing randomization parameters).⁶ However, since such long associations cannot be discovered, in practice, we will not get privacy breaches above 50%.

Summary Despite choosing a conservative privacy breach level of 50%, and further choosing a minimum support around the lowest discoverable support, we were able to successfully find most of the frequent itemsets, with relatively small numbers of false drops and false positives.

⁵If there are no frequent itemsets for some combination, we pick the itemsets with the highest support.

⁶While we could have easily corrected the slip, we felt it more instructive to leave it in.

(a) mailorder, 0.2% minimum support

Itemset Size	True Itemsets	True Positives	False Drops	False Positives
1	65	65	0	0
2	228	212	16	28
3	22	18	4	5

(b) soccer, 0.2% minimum support

Itemset Size	True Itemsets	True Positives	False Drops	False Positives
1	266	254	12	31
2	217	195	22	45
3	48	43	5	26

Table 1: Results on Real Datasets

(a) mailorder, $\geq 0.2\%$ true support

size	Itemsets	predicted support			
		< 0.1	0.1 ± 0.15	0.15 ± 0.2	≥ 0.2
1	65	0	0	0	65
2	228	0	1	15	212
3	22	0	1	3	18

(b) soccer, $\geq 0.2\%$ true support

size	Itemsets	predicted support			
		< 0.1	0.1 ± 0.15	0.15 ± 0.2	≥ 0.2
1	266	0	2	10	254
2	217	0	5	17	195
3	48	0	1	4	43

Table 2: Analysis of false drops

(a) mailorder, $\geq 0.2\%$ predicted support

size	Itemsets	true support			
		< 0.1	0.1 ± 0.15	0.15 ± 0.2	≥ 0.2
1	65	0	0	0	65
2	240	0	0	28	212
3	23	1	2	2	18

(b) soccer, $\geq 0.2\%$ predicted support

size	Itemsets	true support			
		< 0.1	0.1 ± 0.15	0.15 ± 0.2	≥ 0.2
1	285	0	7	24	254
2	240	7	10	28	195
3	69	5	13	8	43

Table 3: Analysis of false positives

soccer										
Transaction size:	1	2	3	4	5	6	7	8	9	10
Worst Average:	1: 4.4%	2: 20.2%	3: 39.2%	4: 44.5%	4: 43.9%	4: 37.5%	4: 36.2%	4: 38.7%	8: 51.0%	10: 49.4%
Worst of the Worst:	1: 45.5%	2: 45.4%	3: 53.2%	4: 49.8%	5: 49.7%	5: 42.7%	5: 41.8%	5: 44.5%	9: 66.2%	10: 65.6%

mailorder						
Transaction size:	1	2	3	4	5	6
Worst Average:	1: 12.0%	2: 27.5%	3: 48.4%	4: 51.5%	5: 51.7%	5: 51.9%
Worst of the Worst:	1: 47.6%	2: 51.9%	3: 53.6%	4: 53.1%	5: 53.6%	6: 55.4%

Table 4: Actual Privacy Breaches

6. CONCLUSIONS

In this paper, we have presented three key contributions toward mining association rules while preserving privacy. First, we pointed out the problem of privacy breaches, presented their formal definitions and proposed a natural solution. Second, we gave a sound mathematical treatment for a class of randomization algorithms and derived formulae for support and variance prediction, and showed how to incorporate these formulae into mining algorithms. Finally, we presented experimental results that validated the algorithm in practice by applying it to two real datasets from different domains.

We conclude by raising three interesting questions for future research. Our approach deals with a restricted (albeit important) class of privacy breaches; can we extend it to cover other kinds of breaches? Second, what are the theoretical limits on discoverability for a given level of privacy (and vice versa)? Finally, can we combine randomization and cryptographic protocols to get the strengths of both without the weaknesses of either?

7. REFERENCES

- [1] N. R. Adam and J. C. Wortman. Security-control methods for statistical databases. *ACM Computing Surveys*, 21(4):515–556, Dec. 1989.
- [2] D. Agrawal and C. C. Aggarwal. On the Design and Quantification of Privacy Preserving Data Mining Algorithms. In *Proc. of the 20th ACM Symposium on Principles of Database Systems*, pages 247–255, Santa Barbara, California, May 2001.
- [3] R. Agrawal. Data Mining: Crossing the Chasm. In *5th Int'l Conference on Knowledge Discovery in Databases and Data Mining*, San Diego, California, August 1999. Available from <http://www.almaden.ibm.com/cs/quest/papers/kdd99.chasm.ppt>.
- [4] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. of the ACM SIGMOD Conference on Management of Data*, pages 207–216, Washington, D.C., May 1993.
- [5] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast Discovery of Association Rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, chapter 12, pages 307–328. AAAI/MIT Press, 1996.
- [6] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules. Research Report RJ 9839, IBM Almaden Research Center, San Jose, California, June 1994.
- [7] R. Agrawal and R. Srikant. Privacy preserving data mining. In *Proc. of the ACM SIGMOD Conference on Management of Data*, pages 439–450, Dallas, Texas, May 2000.
- [8] R. Bayardo. Efficiently mining long patterns from databases. In *Proc. of the ACM SIGMOD Conference on Management of Data*, Seattle, Washington, 1998.
- [9] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, Belmont, 1984.
- [10] Business Week. *Privacy on the Net*, March 2000.
- [11] C. Clifton and D. Marks. Security and privacy implications of data mining. In *ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pages 15–19, May 1996.
- [12] R. Conway and D. Strip. Selective partial access to a database. In *Proc. ACM Annual Conf.*, pages 85–89, 1976.
- [13] L. Cranor, J. Reagle, and M. Ackerman. Beyond concern: Understanding net users' attitudes about online privacy. Technical Report TR 99.4.3, AT&T Labs–Research, April 1999.
- [14] L. F. Cranor, editor. *Special Issue on Internet Privacy*. Comm. ACM, 42(2), Feb. 1999.
- [15] The Economist. *The End of Privacy*, May 1999.
- [16] V. Estivill-Castro and L. Brankovic. Data swapping: Balancing privacy against precision in mining for logic rules. In M. Mohania and A. Tjoa, editors, *Data Warehousing and Knowledge Discovery DaWaK-99*, pages 389–398. Springer-Verlag Lecture Notes in Computer Science 1676, 1999.
- [17] European Union. *Directive on Privacy Protection*, October 1998.
- [18] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *CRYPTO*, pages 36–54, 2000.
- [19] T. M. Mitchell. *Machine Learning*, chapter 6. McGraw-Hill, 1997.
- [20] Office of the Information and Privacy Commissioner, Ontario. *Data Mining: Staking a Claim on Your Privacy*, January 1998.
- [21] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1986.
- [22] A. Shoshani. Statistical databases: Characteristics, problems and some solutions. In *VLDB*, pages 208–213, Mexico City, Mexico, September 1982.
- [23] K. Thearling. Data mining and privacy: A conflict in making. *DS**, March 1998.
- [24] Time. *The Death of Privacy*, August 1997.
- [25] J. Vaidya and C. W. Clifton. Privacy preserving

association rule mining in vertically partitioned data. In *Proc. of the 8th ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, Edmonton, Canada, July 2002.

- [26] S. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *J. Am. Stat. Assoc.*, 60(309):63–69, March 1965.
- [27] A. Westin. E-commerce and privacy: What net users want. Technical report, Louis Harris & Associates, June 1998.
- [28] A. Westin. Privacy concerns & consumer choice. Technical report, Louis Harris & Associates, Dec. 1998.
- [29] A. Westin. Freebies and privacy: What net users think. Technical report, Opinion Research Corporation, July 1999.

APPENDIX

A. PROOFS

A.1 Proof of Statement 1

PROOF. Each coordinate $N \cdot s'_{l'}$ of the vector in (4) is, by definition of partial supports, just the number of transactions in the randomized sequence T' that have intersections with A of size l' . Each randomized transaction t' contributes to one and only one coordinate $N \cdot s'_{l'}$, namely to the one with $l' = \#(t' \cap A)$. Since we are dealing with a per-transaction randomization, different randomized transactions contribute independently to one of the coordinates. Moreover, by item-invariance assumption, the probability that a given randomized transaction contributes to the coordinate number l' depends only on the size of the original transaction t (which equals m) and the size l of intersection $t \cap A$. This probability equals $p[l \rightarrow l']$.

So, for all transactions in T that have intersections with A of the same size l (and there are $N \cdot s_l$ such transactions) the probabilities of contributing to various coordinates $N \cdot s'_{l'}$ are the same. We can split all N transactions into $k+1$ groups according to their intersection size with A . Each group contributes to the vector in (4) as a multinomial distribution with probabilities

$$(p[l \rightarrow 0], p[l \rightarrow 1], \dots, p[l \rightarrow k]),$$

independently from the other groups. Therefore the vector in (4) is a sum of $k+1$ independent multinomials. Now it is easy to compute both expectation and covariance.

For a multinomial distribution (X_0, X_1, \dots, X_k) with probabilities (p_0, p_1, \dots, p_k) , where $X_0 + X_1 + \dots + X_k = n$, we have $\mathbf{E} X_i = n \cdot p_i$ and

$$\mathbf{Cov}(X_i, X_j) = \mathbf{E}(X_i \Leftrightarrow p_i)(X_j \Leftrightarrow p_j) = n \cdot (p_i \delta_{i=j} \Leftrightarrow p_i p_j).$$

In our case, $X_i = l'$'s part of $N \cdot s'_{l'}$, $n = N \cdot s_l$, and $p_i = p[l \rightarrow i]$. For a sum of independent multinomial distri-

butions, their expectations and covariances add together:

$$\begin{aligned} \mathbf{E}(N \cdot s'_{l'}) &= \sum_{l=0}^k N \cdot s_l \cdot p[l \rightarrow l'], \\ \mathbf{Cov}(N \cdot s'_i, N \cdot s'_j) &= \\ &= \sum_{l=0}^k N \cdot s_l \cdot (p[l \rightarrow i] \cdot \delta_{i=j} \Leftrightarrow p[l \rightarrow i] \cdot p[l \rightarrow j]) \end{aligned}$$

Thus, after dividing by an appropriate power of N , the formulae in the statement are proven. \square

A.2 Proof of Statement 2

PROOF. We are given a transaction $t \in T$ and an itemset $A \subseteq \mathcal{I}$, such that $|t| = m$, $|A| = k$, and $\#(t \cap A) = l$. In the beginning of randomization, a number j is selected with distribution $\{p_m[j]\}$, and this is what the first summation takes care of. Now assume that we retain exactly j items of t , and discard $m \Leftrightarrow j$ items.

Suppose there are q items from $t \cap A$ among the retained items. How likely is this? Well, there are $\binom{m}{j}$ possible ways to choose j items from transaction t ; and there are $\binom{l}{q} \binom{m-l}{j-q}$ possible ways to choose q items from $t \cap A$ and $j \Leftrightarrow q$ items from $t \setminus A$. Since all choices are equiprobable, we get $\binom{l}{q} \binom{m-l}{j-q} / \binom{m}{j}$ as the probability that exactly q A -items are retained.

To make t' contain exactly l' items from A , we have to get additional $l' \Leftrightarrow q$ items from $A \setminus t$. We know that $\#(A \setminus t) = k \Leftrightarrow l$, and that any such item has probability ρ to get into t' . The last terms in (8) immediately follow. Summation bounds restrict q to its actually possible (= nonzero probability) values. \square

A.3 Proof of Statement 3

PROOF. Let us denote

$$\begin{aligned} \vec{p}_l &:= (p[l \rightarrow 0], p[l \rightarrow 1], \dots, p[l \rightarrow k])^T, \\ \vec{q}_l &:= (q[l \leftarrow 0], q[l \leftarrow 1], \dots, q[l \leftarrow k])^T. \end{aligned}$$

Since $PQ = QP = I$ (where I is the identity matrix), we have

$$\sum_{l=0}^k p[l \rightarrow i] q[l \leftarrow j] = \sum_{l'=0}^k p[i \rightarrow l'] q[j \leftarrow l'] = \delta_{i=j}.$$

Notice also, from (7), that matrix $D[l]$ can be written as

$$D[l] = \text{diag}(\vec{p}_l) \Leftrightarrow \vec{p}_l \vec{p}_l^T,$$

where $\text{diag}(\vec{p}_l)$ denotes the diagonal matrix with \vec{p}_l -coord-

inates as its diagonal elements. Now it is easy to see that

$$\begin{aligned}
\tilde{s} &= \tilde{q}_k^T \tilde{s}' = \sum_{l'=0}^k q[k \leftarrow l'] \cdot s_{l'}; \\
\mathbf{Var} \tilde{s} &= \frac{1}{N} \sum_{l=0}^k s_l \tilde{q}_k^T D[l] \tilde{q}_k = \\
&= \frac{1}{N} \sum_{l=0}^k s_l \tilde{q}_k^T (\text{diag}(\tilde{p}_l) \Leftrightarrow \tilde{p}_l \tilde{p}_l^T) \tilde{q}_k = \\
&= \frac{1}{N} \sum_{l=0}^k s_l (\tilde{q}_k^T \text{diag}(\tilde{p}_l) \tilde{q}_k \Leftrightarrow (\tilde{p}_l^T \tilde{q}_k)^2) = \\
&= \frac{1}{N} \sum_{l=0}^k s_l \left(\sum_{l'=0}^k p[l \rightarrow l'] q[k \leftarrow l']^2 \Leftrightarrow \delta_{l=k} \right); \\
(\mathbf{Var} \tilde{s})_{\text{est}} &= \\
&= \frac{1}{N} \sum_{l=0}^k (\tilde{q}_l^T \tilde{s}') \left(\sum_{l'=0}^k p[l \rightarrow l'] q[k \leftarrow l']^2 \Leftrightarrow \delta_{l=k} \right) = \\
&= \frac{1}{N} \sum_{j=0}^k s'_j \left(\sum_{l, l'=0}^k q[l \leftarrow j] p[l \rightarrow l'] q[k \leftarrow l']^2 \Leftrightarrow \right. \\
&\Leftrightarrow \sum_{l=0}^k \delta_{l=k} q[l \leftarrow j] \left. \right) = \frac{1}{N} \sum_{j=0}^k s'_j \left(\sum_{l'=0}^k \delta_{l'=j} q[k \leftarrow l']^2 \Leftrightarrow \right. \\
&\Leftrightarrow q[k \leftarrow j] \left. \right) = \frac{1}{N} \sum_{j=0}^k s'_j (q[k \leftarrow j]^2 \Leftrightarrow q[k \leftarrow j]).
\end{aligned}$$

□

A.4 Proof of Statement 4

PROOF. We prove the left formula in (13) first, and then show that the right one follows from the left one. Consider $N \cdot \Sigma_l$; it equals

$$\begin{aligned}
N \cdot \Sigma_l &= N \cdot \sum_{C \subseteq A, |C|=l} \text{supp}^T(C) = \sum_{C \subseteq A, |C|=l} \# \{t_i \in T \mid C \subseteq t_i\} = \\
&= \sum_{i=1}^N \# \{C \subseteq A \mid |C|=l, C \subseteq t_i\}.
\end{aligned}$$

In other words, each transaction t_i should be counted as many times as many different l -sized subsets $C \subseteq A$ it contains. From simple combinatorics we know that if $j = \#(A \cap t_i)$ and $j \geq l$, then t_i contains $\binom{j}{l}$ different l -sized subsets of A . Therefore,

$$\begin{aligned}
N \cdot \Sigma_l &= \sum_{i=1}^N \binom{\#(A \cap t_i)}{l} = \\
&= \sum_{j=l}^k \binom{j}{l} \cdot \# \{t_i \in T \mid \#(A \cap t_i) = j\} = \sum_{j=l}^k \binom{j}{l} N \cdot s_j,
\end{aligned}$$

and the left formula is proven. Now we can check the right formula just by replacing the Σ_j 's according to the left for-

mula. We have:

$$\begin{aligned}
\sum_{j=l}^k (\Leftrightarrow 1)^{j-l} \binom{j}{l} \Sigma_j &= \sum_{j=l}^k (\Leftrightarrow 1)^{j-l} \binom{j}{l} \sum_{q=j}^k \binom{q}{j} s_q = \\
&= \sum_{l \leq j \leq q \leq k} (\Leftrightarrow 1)^{j-l} \binom{j}{l} \binom{q}{j} s_q = \sum_{q=l}^k s_q \sum_{j=l}^q (\Leftrightarrow 1)^{j-l} \binom{j}{l} \binom{q}{j} = \\
&= \sum_{q=l}^k s_q \sum_{j'=0}^{q-l} (\Leftrightarrow 1)^{j'} \frac{(j'+l)!}{l! j'!} \frac{q!}{(j'+l)! (q \Leftrightarrow j' \Leftrightarrow l)!} = \\
&= \sum_{q=l}^k s_q \cdot \frac{q!}{l! (q \Leftrightarrow l)!} \sum_{j'=0}^{q-l} (\Leftrightarrow 1)^{j'} \frac{(q \Leftrightarrow l)!}{j'! (q \Leftrightarrow l \Leftrightarrow j')!} = \\
&= \sum_{q=l}^k s_q \binom{q}{l} \sum_{j'=0}^{q-l} (\Leftrightarrow 1)^{j'} \binom{q \Leftrightarrow l}{j'} = s_l,
\end{aligned}$$

since the sum $\sum_{j'=0}^{q-l} (\Leftrightarrow 1)^{j'} \binom{q \Leftrightarrow l}{j'}$ is zero whenever $q \Leftrightarrow l > 0$.

To prove that matrix P becomes lower triangular after the transformation from \tilde{s} and \tilde{s}' to $\tilde{\Sigma}$ and $\tilde{\Sigma}'$, let us find how $\mathbf{E} \tilde{\Sigma}'$ depends on $\tilde{\Sigma}$ using the definition (12).

$$\begin{aligned}
\mathbf{E} \Sigma_{l'}' &= \sum_{C \subseteq A, |C|=l'} \mathbf{E} \text{supp}^{T'}(C) = \\
&= \sum_{C \subseteq A, |C|=l'} \sum_{l=0}^{l'} p_{l'}^m[l \rightarrow l'] \cdot \text{supp}_l^T(C) = \\
&= \sum_{C \subseteq A, |C|=l'} \sum_{l=0}^{l'} p_{l'}^m[l \rightarrow l'] \sum_{j=l}^{l'} (\Leftrightarrow 1)^{j-l} \binom{j}{l} \Sigma_j(C, T) = \\
&= \sum_{j=0}^{l'} \sum_{l=0}^j \underbrace{(\Leftrightarrow 1)^{j-l} \binom{j}{l} p_{l'}^m[l \rightarrow l']}_{c_{l'j}} \sum_{C \subseteq A, |C|=l'} \Sigma_j(C, T) = \\
&= \sum_{j=0}^{l'} c_{l'j} \sum_{C \subseteq A, |C|=l'} \sum_{B \subseteq C, |B|=j} \text{supp}^T(B) = \\
&= \sum_{j=0}^{l'} c_{l'j} \sum_{B \subseteq A, |B|=j} \# \{C \mid B \subseteq C \subseteq A, |C|=l'\} \cdot \text{supp}^T(B) = \\
&= \sum_{j=0}^{l'} c_{l'j} \sum_{B \subseteq A, |B|=j} \binom{k \Leftrightarrow j}{l' \Leftrightarrow j} \text{supp}^T(B) = \sum_{j=0}^{l'} c_{l'j} \binom{k \Leftrightarrow j}{l' \Leftrightarrow j} \cdot \Sigma_j.
\end{aligned}$$

Now it is clear that only the lower triangle of the matrix can have non-zeros. □

EXHIBIT C

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents

P.O. BOX 1450

Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.132

I, Johannes Gehrke, hereby declare the following:

[0001] I am a co-author with Alexandre Evfimievski, Ramakrishnan Srikant and Rakesh Agrawal on the following paper:

A. Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, "Privacy Preserving Mining of Association Rules," Proc. Of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002, referred to herein as "Privacy Preserving Mining of Association Rules" (July 2002).

[0002] I have read U.S. Patent Application Serial No. 10/624,069, including claims 1-24.

[0003] "Privacy Preserving Mining of Association Rules" (July 2002) discusses the invention that is defined by claims 1-24 of U.S. Patent Application Serial No. 10/624,069.

[0005] I understand that Alexandre Evfimievski, Ramakrishnan Srikant, and Rakesh Agrawal are joint inventors of the invention that is defined by claims 1-24 of U.S. Patent Application Serial No. 10/624,069.

[0006] Although I am a co-author of "Privacy Preserving Mining of Association Rules" (July 2002), I am not an inventor of the invention that is defined by claims 1-24 of U.S. Patent Application Serial No. 10/624,069.

[0007] The above declarations are made according to the best of my recollection upon review of the appropriate documents and notes, and I hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the application or any patent issuing thereon. All statements that are made herein of my own knowledge are true and all statements that are made herein based on information and belief are believed to be true.


Johannes Gehrke

4/10/2006
(Date)



International Business Machines Corporation
Research Division, Almaden Research Center
Intellectual Property Law Department

Marc D. McSwain
650 Harry Road, C4TA/J2B
San Jose, CA 95120-6099
(408) 927-3364
(408) 927-3375 - Fax
mmcswain@us.ibm.com

May 15, 2003

Frederick W. Gibb III, Esq.
McGinn & Gibb, PLLC

Re: new docket ARC920030034US1

Hi Fred:

Please prepare and file a new patent application for this invention; a disclosure and a journal article describing the invention are attached. The article is available online at:
<http://www.almaden.ibm.com/cs/people/srikant/papers/kdd02.pdf>
and is generally cited as:

A. Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, "Privacy Preserving Mining of Association Rules," Proc. of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002.

The primary contact inventor is Ramakrishnan Srikant, who is at (408)927-1774, (408)927-3215 fax, and srikant@almaden.ibm.com. The other inventors are Rakesh Agrawal and Alexandre Evfimiefski, who was a summer student IBM employee here when the invention was conceived. Professor Gehrke is listed as an author on the article, but is not an inventor.

This invention was disclosed on July 23, 2002, so there's a statutory bar date approaching. We'll send you the references cited in the journal article and full inventor information separately. This invention is quite mathematical, so I would suggest placing the proofs into an Appendix. Dr. Srikant can help with reformatting the journal article so you can re-use as much of his material as directly as possible.

Thanks,

Marc

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

**PETITION UNDER 37 C.F.R. §1.147(a) TO PERMIT ACCEPTANCE OF 37
C.F.R. AFFIDAVIT WITHOUT THE SIGNATURE OF ONE OF THE
INVENTORS**

Director of the USPTO
P.O. Box 1450
Alexandria, VA 22313-1450
Sir:

Applicants hereby petition the Director to accept the 37 C.F.R. §1.131 declaration submitted as ATTACHMENT A with the simultaneously filed submission in support of the request for continued examination under 37 C.F.R. §1.114. The 37 C.F.R. §1.131 declaration has been signed by two of the joint inventors of the present invention, namely, Alexandre Evfimievski and Ramakrishnan Srikant. Another of the joint inventors (Rakesh Agrawal) was at the time of the invention and at the time the present patent application was filed, an employee of the assignee, International Business Machines, Inc. However, Rakesh Agrawal no longer works for International Business Machines, Inc. and has refused to sign a 37 C.F.R. §1.131 declaration, which is necessary to overcome a prior art rejection.

Per the requirements of 37 C.F.R. §1.147(a), the Applicants also submit that the last known work and home addresses of the non-signing inventor, Rakesh Aggrawal, was as follows:

Work:
Microsoft Research Labs
1065 La Avenida
Mountain View, CA 94043

Home:
1290 Quail Creek Circle
San Jose, CA 95120

Per the requirements of 37 C.F.R. §1.147(a), the

Applicants provide the attached statement (ATTACHMENT (a)) of Van Nguy, an attorney at the IBM Almaden Research Center in San Jose, California, along with copies of email correspondence between Van Nguy and Rakesh Aggrawal (EXHIBITS (1)-(16) to Attachment (a)). The statement and emails show Van Nguy presented Rakesh Aggrawal with three different versions of a 37 C.F.R. §1.131 declaration for his signature and that he refused to sign all three of the different versions.

Also provided in support of Van Nguy's statement are copies of the three different versions of the 37 C.F.R. §1.131 declaration presented to Rakesh Aggrawal by Van Nguy for his signature (see Attachments (b)-(d)). Attachment (b) differed from the executed declaration of Alexandre Evfimievski and Ramakrishnan Srikant only slightly. For example, Attachment (b) used the phrase "earliest effective prior art date", rather than the actual August 2002 date of Rizvi. Additionally, Attachment (b) contained the

statement “During all time periods mentioned herein and, specifically, between the conception date and the filing date of the Patent Application, all activities described herein occurred in the United States.”

Attachments (c) and (d) differed from the executed declaration of Alexandre Evfimievski and Ramakrishnan Srikant in that they were significantly shorter per Rakesh Agrawal’s request. For example, Attachments (c) and (d) did not contain the listing of independent claims with reference to the exemplary locations within “Privacy Preserving Mining of Association Rules” (July 2002) (i.e., the published article of the inventors upon which the present application was based), wherein the claimed features are described. This listing was provided in the original version of the declaration presented to Rakesh Agrawal (i.e., Attachment (a)) and was retained in the executed declaration of Alexandre Evfimievski and Ramakrishnan Srikant.

The Applicants submit that the statement of Van Nguy, as well as the other evidence provided, clearly indicate that Rakesh Agrawal is unwilling to or unable to continue to join in the prosecution of the present application. His refusal to sign a 37 C.F.R. §1.131 affidavit was not based on a difference of opinion as to the facts of the case, but rather on Rakesh Agrawal’s misunderstanding of the requirements of a 37 C.F.R. §1.131 affidavit and, more specifically, his belief, despite efforts to persuade him otherwise, that a statement claiming that “our invention was done before July 2002” would be sufficient. Therefore, the Applicants respectfully request that the executed declaration of Alexandre Evfimievski and Ramakrishnan Srikant without the signature of

Rakesh Agrawal be accepted for purposes of establishing prior invention under 37 C.F.R. §1.131, as failure to do so will result in irreparable damage to the remaining inventors.

Please charge the petition fee under §1.17(g) of \$200.00 for a §1.147 petition and any other fees required to Attorney's Deposit Account No. 09-0441.

Respectfully submitted,

Dated: 7/3/08

/Pamela M. Riley/
Pamela M. Riley
Registration No. 40,146

Gibb & Rahman, LLC
2568-A Riva Road, Suite 304
Annapolis, MD 21401
Voice: (410) 573-0227
Fax: (301) 261-8825
Customer Number: 29154

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION IN SUPPORT OF 37 C.F.R. 1.147(a) PETITION

I, Van Nguy, an attorney with IBM, located at IBM Almaden Research Center, 650 Harry Road, C4TA/J2B, San Jose, CA 95120, hereby declare the following:

International Business Machines Corporation (IBM) is the assignee of the above-referenced patent application (herein after referred to as the Patent Application). Rakesh Agrawal is one of the joint inventors on the Patent Application. At the time of the invention that is the subject matter of the Patent Application and further at the time the Patent Application was filed, Rakesh Agrawal was an employee of IBM. Rakesh Agrawal is no longer employed by IBM.

On May 28, 2008, I personally presented Rakesh Agrawal, via email, with an original version of a 37 C.F.R. §1.131 declaration, regarding the above-referenced patent application, for his signature. He refused to sign this original version. Specifically, over the course of several emails, he indicated that he was only willing to sign a “declaration that our invention was done before July 2002” and he questioned the use of terms of art

such as “earliest effective prior art date.” I tried to explain to him the purpose of the declaration, that all inventors are required to sign the declaration and that the limited statement he proposed may be insufficient.

On June 3, 2008, after Rakesh Agrawal refused to sign the original version of the declaration, I presented him with second shorter of the declaration. Specifically, in this second version the listing of the independent claims along with the subsections of the July 2002 paper of the Applicants which disclosed the claimed features was eliminated. Rakesh Agrawal had questions regarding the language used in this second version and, therefore, on that same day I presented him with a third version of the declaration. Specifically, in this third version the phrase “earliest effective prior art date” was replaced with August 2002. Rakesh Agrawal continued to refuse to sign the declaration and indicated that he had “spent way to much time iterating” with me on it.

The following is a timeline and a summary of the email correspondence I have had with Rakesh Agrawal regarding the 37 C.F.R. §1.131 declaration:

May 15, 2008--I emailed Rakesh Agrawal, asking if he would be “open to receiving and signing the declaration” (Exhibit 1).

May 22, 2008--As I had received no response from Rakesh Agrawal, I again emailed him asking if he would be “open to receiving and signing the declaration” (Exhibit 2).

May 22, 2008—Rakesh Agrawal responded, asking the meaning of “you and your co-inventors invented before Rizvi” (Exhibit 3).

May 23, 2008—I responded to Rakesh Agrawal, answering his question and again asking if he was “open to receiving the declaration for signature” (Exhibit 4).

May 24, 2008—Rakesh Agrawal responded, asking “Isn’t it obvious that our invention was done before August 2002 since the SIGKDD paper containing the invention was published in July 2002?” (Exhibit 5).

May 27, 2008—I responded to Rakesh Agrawal, answering his question and informing him that I would send him the declaration for his review and signature (Exhibit 6).

May 27, 2008—Rakesh Agrawal responded, stating “I can sign the declaration that our was done before July 2002” (Exhibit 7).

May 28, 2008—I forwarded the original version of the declaration to Rakesh Agrawal for his signature and review (Exhibit 8).

May 28, 2008—Rakesh Agrawal responded, indicating “I thought I was simply declaring that: Ok, I can sign the declaration that our invention was done before July 2002” (Exhibit 9).

May 28, 2008—I responded to Rakesh Agrawal, explaining the contents of the declaration (Exhibit 10).

May 28, 2008—Rakesh Agrawal responded, reiterating that he would only sign the declaration “that our invention was done before July 2002” (Exhibit 11).

May 28, 2008—I responded to Rakesh Agrawal, again explaining the contents of the declaration and informing him that the statement he proposed may be insufficient (Exhibit 12).

June 3, 2008—I forwarded a second shorter version of the declaration to Rakesh Agrawal (Exhibit 13), as discussed above.

June 3, 2008—Rakesh Agrawal responded to receipt of the second version, questioning the use of terms of art such as “earliest effective prior art date” and again indicated that he would not be willing to sign anything beyond “when we did the invention” (Exhibit 14).

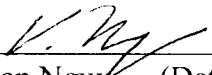
June 3, 2008—I responded to Rakesh Agrawal, sending him a third version of the declaration that replaced the phrase “earliest effective prior art date” with August 2002 (Exhibit 15), as discussed above.

June 3, 2008—Rakesh Agrawal responded stating, “I’m sorry I cannot sign this version. And frankly, I have spent way too much time iterating with you on it.” (Exhibit 16).

I did not submit any additional versions of the declaration to Rakesh Agrawal for his signature, as he indicated that he was not receptive to receiving any version of the declaration from me that went beyond the statement “our invention was done before July 2002.” Furthermore, since 37 C.F.R. §1.131 requires a declaration that shows either reduction to practice prior to the effective date of the reference or conception of the invention prior to the effective date of the reference coupled with due diligence from prior to that date to the filing of the application, I believed that a declaration solely stating “our invention was done before July 2002” would be insufficient to remove the Rizvi article as a prior art reference. I believe Mr. Agrawal’s refusal to sign is based upon not understanding the legal reasons for submitting a declaration (e.g., see Exhibit 5), upon not understanding legal terms of art within the declaration (e.g., see Exhibit 14), and upon not understanding the factors that are weighed in determining whether a declaration is sufficient to remove a reference as prior art. I am not in a position to explain details of

these matters to Mr. Agrawal as he no longer works for IBM and as I do not represent him as an individual.

The above statements are made according to the best of my recollection, upon review of the appropriate documents and notes. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true. I further hereby acknowledge that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and that such willful false statements may jeopardize the validity of the Patent Application or any patent issued thereon.

 7/1/08

Van Nguy (Date)

From: Van Nguy/Almaden/IBM
To: rakesh.devnull@[REDACTED], rakesh.prof@[REDACTED], ragrawal@[REDACTED]
rakesh.agrawal@[REDACTED]
Date: 05/15/2008 03:09 PM
Subject: *IBM Confidential: Signature needed to declare that you invented first

Hi Rakesh -

Re: US patent application 10/624,069 (Publication No 20050021488A1)
Entitled: Mining Association Rules over Privacy Preserving Data

The above case is still under examination at the United States Patent and Trademark Office (USPTO). The USPTO has cited an article (Rizvi et al) that was published before the filing of the above application, but after your article at ACM SIGKDD 2002 describing your invention. Rizvi actually cites your article as a reference.

To confirm to the Patent Office that you and your co-inventors invented before Rizvi, we must file a declaration signed by all inventors declaring that you invented first. We would like to prepare the declaration for your signature. We wanted to make contact with you first to make sure that you are open to receiving and signing the declaration.

Please let us know by THURSDAY MAY 22, 2008 if you approve us sending you the declaration for signature.

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Van Nguy/Almaden/IBM
To: ragrawal@[REDACTED], rakesh.agrawal@[REDACTED], rakesh.devnull@[REDACTED], rakesh.prof@[REDACTED]
Date: 05/22/2008 03:42 PM
Subject: Re: Signature needed to declare that you invented first

Hi Rakesh -

I wanted to follow up on the email below since I have not heard from you. Are you open to us sending you a declaration to sign regarding the below?

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Van Nguy/Almaden/IBM
To: rakesh.devnull@[REDACTED], rakesh.prof@[REDACTED], ragrawal@[REDACTED], rakesh.agrawal@[REDACTED]
Date: 05/15/2008 03:09 PM
Subject: *IBM Confidential: Signature needed to declare that you invented first

Hi Rakesh -

Re: US patent application 10/624,069 (Publication No 20050021488A1)
Entitled: Mining Association Rules over Privacy Preserving Data

The above case is still under examination at the United States Patent and Trademark Office (USPTO). The USPTO has cited an article (Rizvi et al) that was published before the filing of the above application, but after your article at ACM SIGKDD 2002 describing your invention. Rizvi actually cites your article as a reference.

To confirm to the Patent Office that you and your co-inventors invented before Rizvi, we must file a declaration signed by all inventors declaring that you invented first. We would like to prepare the declaration for your signature. We wanted to make contact with you first to make sure that you are open to receiving and signing the declaration.

Please let us know by THURSDAY MAY 22, 2008 if you approve us sending you the declaration for signature.

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120

[REDACTED]

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: Van Nguy/Almaden/IBM@IBMUS
Cc: [REDACTED]
Date: 05/22/2008 11:10 PM
Subject: RE: Signature needed to declare that you invented first

you and your co-inventors invented before Rizvi --- what does it mean?

From: Van Nguy [REDACTED]
Sent: Friday, May 23, 2008 9:31 AM
To: Rakesh Agrawal
Cc: [REDACTED]
Subject: RE: Signature needed to declare that you invented first

Hi Rakesh -

Thank you for your response.

It means that the Patent Office thinks that the attached article by Rizvi et al. ("the Rizvi article") is "prior art" to your invention because the Rizvi article was published in August 2002 and your patent application (entitled "Mining Association Rules Over Privacy Preserving Data") was filed in July 2003. However, in the US, if we can show that you and your co-inventors invented before the effective date of the reference (here, August 20, 2002), then the reference is not really "prior art."

One of the evidence we have that the Rizvi article is not prior art to your invention is a paper entitled "Privacy Preserving Mining of Association Rules" by you, your co-inventors (Alexandre Evfimievski and Ramakrishan Srikant) and Johannes Gehrke that includes a description of your the invention (as confirmed by Alexandre Evfimievski) and was published at ACM SIGKDD **July 2002**. The Rizvi article actually cites your article as reference [8].

The Patent Office requires, however, that to submit this evidence and to establish invention prior to the effective date of the reference (thus removing the Rizvi article as a "prior art"), we must submit a declaration signed by **all** the inventors saying as such.

So, if you are open to receiving the declaration for signature, we will go ahead and prepare the document and send to you (which we can do over email) for your signature. Please let us know if you are open to this.

(See attached file: p682-rizvi.pdf)(See attached file: p217-evfimievski.pdf)

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

Rakesh Agrawal

<Rakesh.Agrawal@[REDACTED]>

05/24/2008 02:22 PM

To Van Nguy/Almaden/IBM@IBMUS

cc "[REDACTED]"

Subject RE: Signature needed to declare that you invented first

Isn't it obvious that our invention was done before August 2002 since the SIGKDD paper containing the invention was published in July 2002?

From: Van Nguy/Almaden/IBM
To: [REDACTED], Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
Date: 05/27/2008 09:12 AM
Subject: RE: Signature needed to declare that you invented first

Hi Rakesh -

Notwithstanding that, we cannot submit the evidence to the USPTO unless we have a declaration signed by all the inventors. Based on the emails below, I will assume that you are willing to receive the declaration for review and signature. I will go ahead and have it prepared. I will forward it onto you as soon as it is ready.

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: [REDACTED]
Cc: Van Nguy/Almaden/IBM@IBMUS
Date: 05/27/2008 10:26 AM
Subject: RE: Signature needed to declare that you invented first

Ok, I can sign the declaration that our was done before July 2002.

From: Van Nguy [REDACTED]
Sent: Wednesday, May 28, 2008 8:28 AM
To: Rakesh Agrawal
Cc: [REDACTED]
Subject: Fw: Signature needed to declare that you invented first

Rakesh - Please find the attached for your review. Please sign the declaration and postal mail, email, or fax the original back to me at the number below. Best regards, Van

(See attached file: Rizvi article.pdf)(See attached file: ALM 5074 1.131 declaration.doc)(See attached file: ARC920030034USI_Attachment C.pdf)(See attached file: privacy preserving mining of association rules.pdf)

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: Van Nguy/Almaden/IBM@IBMUS
Cc: [REDACTED]
Date: 05/28/2008 08:42 AM
Subject: RE: Signature needed to declare that you invented first

I thought I was simply declaring that:

Ok, I can sign the declaration that our invention was done before July 2002.

From: Van Nguy [REDACTED]
Sent: Wednesday, May 28, 2008 1:22 PM
To: Rakesh Agrawal
Cc: [REDACTED]
Subject: RE: Signature needed to declare that you invented first

Hi Rakesh -

The declaration provides a detailed description of the evidence as they directly relate to the current claims since your "invention" is legally defined by the claims, though described, supported, and enabled by the specification. For your convenience, I have attached the declaration again here for your signature and return by postal mail, email, or fax to us.

(See attached file: ALM 5074 1.131 declaration.doc)

We look forward to hearing from you.

Best regards,
Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: Van Nguy/Almaden/IBM@IBMUS
Cc: [REDACTED]
Date: 05/28/2008 02:07 PM
Subject: RE: Signature needed to declare that you invented first

Again: As I said earlier, I can sign the declaration that our invention was done before July 2002. The document you are asking me to sign seems to make representations beyond that!!!

From: Van Nguy/Almaden/IBM
To: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
Cc: [REDACTED]
Date: 05/28/2008 04:15 PM
Subject: RE: Signature needed to declare that you invented first

Hi Rakesh - Yes. Unfortunately, the declaration needs to say what "our" and "invention" means in relation to the cited reference, so the declaration specifies this. The blanket statement you propose may be insufficient and we did not want to have to come back to you for another signature. We know you are very busy. Attached again is the declaration for your convenience.

[attachment "ALM 5074 1.131 declaration.doc" deleted by Van Nguy/Almaden/IBM]

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

IBM CONFIDENTIAL

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: Van Nguy/Almaden/IBM@IBMUS
Cc: [REDACTED]
Date: 06/03/2008 12:38 PM
Subject: RE: Signature needed to declare that you invented first

What is meant by - earliest effective prior art date of Rizvi?

I have no way of knowing when Rizvi et al conceived their innovation. There is usually a gap between when the invention is conceived and when a paper containing an invention is published.

I told this to Laura – I can sign a declaration that says when we did the invention. Anything beyond, I am not comfortable.

Also, please don't send anything marked IBM confidential to me.

From: Van Nguy [REDACTED]
Sent: Tuesday, June 03, 2008 1:54 PM
To: Rakesh Agrawal
Cc: [REDACTED]
Subject: RE: Signature needed to declare that you invented first

Hi Rakesh -

I have asked the declaration be amended to remove references to terms of art (e.g., "earliest effective prior art date of Rizvi?") and replaced with an actual date (in this case, August 2002). So paragraph [0001] now begins:

The purpose of this declaration is to prove that we conceived the claimed invention prior to the August 2002 date of **Exhibit A**.

I hope this is a version meets with your approval. Thank you for all your attention to this matter.

(See attached file: REVISED ALM 5074 1 131 declaration short version.doc)

Best regards,

Van N. Nguy
Attorney, IBM Almaden Research Center
650 Harry Road, C4TA/J2B, San Jose, CA 95120
[REDACTED]

PREPARED BY OR FOR IBM ATTORNEY/PRIVILEGE REVIEW REQUIRED: This e-mail may contain privileged information and/or attorney work product, as it was prepared by or for an IBM attorney. Use or disclosure of such information by anyone other than a designated addressee is unauthorized. Do not copy or forward without authorization. If you are not an intended recipient, please notify the sender and delete this e-mail.

EXHIBIT 15

From: Rakesh Agrawal <Rakesh.Agrawal@[REDACTED]>
To: Van Nguy/Almaden/IBM@IBMUS
Cc: [REDACTED]
Date: 06/03/2008 04:47 PM
Subject: RE: Signature needed to declare that you invented first

I'm sorry I cannot sign this version. And frankly, I have spent way too much time iterating with you on it.

EXHIBIT 16

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

We, Alexandre Evfimievski, Ramakrishnan Srikant, and Rakesh Agrawal, the Applicants and joint inventors of the above-referenced invention defined by claims 1-24 and disclosed in U.S. Patent Application Serial No. 10/624,069 hereby declare the following:

[0001] The purpose of this declaration is to prove that we conceived the claimed invention prior to the earliest effective prior art date of **Exhibit A**. Exhibit A is a copy of the following published article cited in the March 5, 2008 rejection of claims 1-24 of the present patent application (herein after referred to as Patent Application) under 35 U.S.C. §102(a): Rizvi, et al., "Maintaining Data Privacy in Association Rule Mining," Proceedings of the 28th VLDB Conference, Hong Kong, China, 12 pages, dated August 2002 (hereinafter referred to as Rizvi).

[0002] The following shows that we conceived our invention prior to the August 2002

earliest effective prior art date of Rizvi, that we were diligent from the date of conception to the date of reduction to practice and that we were further diligent to the date of the filing of the patent application (herein after referred to as Patent Application), which has a filing date of July 21, 2003.

[0003] During all time periods mentioned herein and, specifically, between the conception date and the filing date of the Patent Application, all activities described herein occurred in the United States.

[0004] Proof of the conception of the claimed invention prior to August 2002 and diligence in reducing the invention to practice and filing the Patent Application is demonstrated by the attached **Exhibit B** in conjunction with **Exhibit A**.

[0005] **Exhibit B** is a copy of the following published paper: Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, “Privacy Preserving Mining of Association Rules,” Proc. Of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002, referred to herein as “Privacy Preserving Mining of Association Rules” (July 2002).

[0006] Each of the Applicants of the Patent Application are co-authors on the paper “Privacy Preserving Mining of Association Rules” (July 2002) along with J. Gehrke.

[0007] J. Gehrke was a professor and advisor of A. Evfimievski, during the time period in which the idea for the invention was conceived. Although J. Gehrke is listed as a co-author of

“Privacy Preserving Mining of Association Rules” (July 2002), he was not an inventor of the invention defined by claims 1-24 of the Patent Application.

[0008] J. Gehrke has read the Patent Application and has declared that he is not an inventor of the invention defined by claims 1-24 (**see Exhibit C**). We, the Applicants, also acknowledge that J. Gehrke was not an inventor of the invention defined by claims 1-24 of the Patent Application. Therefore, the portions of “Privacy Preserving Mining of Association Rules” (July 2002), which describe the features of claims 1-24 of the Patent Application, describe the Applicants’ own work and no one else’s.

[0009] “Privacy Preserving Mining of Association Rules” (July 2002) describes the invention defined by claims 1-24. In fact “Privacy Preserving Mining of Association Rules” (July 2002) served as the basis for the specification, drawings and claims of the Patent Application.

[0010] The following is a listing of independent claims 1, 7, 13, and 19 of the Patent Application that define the present invention with reference to the exemplary locations within “Privacy Preserving Mining of Association Rules” (July 2002), wherein the claimed feature is described:

Claim 1: A computer-implemented method of mining association rules over transactions from datasets while maintaining privacy of individual transactions within said datasets through randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see section 4.1], said randomizing comprising:

randomly dropping true items from each transaction in said original dataset[see section 4.1]; and

randomly inserting false items into each transaction in said original dataset[see section 4.1];

collecting said randomized dataset in a database [see section 4.1];

determining support of an association rule in said randomized dataset [see section 4.2];

estimating support of said association rule in said original dataset based on said support of said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said randomizing, privacy breaches of said individual transactions are controlled [see section 4.4].

Claim 7: A computer-implemented method of mining association rules from databases while maintaining privacy of individual transactions within said databases through randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see section 4.1], said randomizing comprising:

randomly dropping true items from each transaction in said original dataset [see section 4.1];

randomly inserting false items into each transaction in said original dataset [see section 4.1];

collecting said randomized dataset in a database [see section 4.1];

mining said database to recover an association rule in said original dataset after said dropping and inserting processes, wherein said mining comprising [see sections 4.2-4.5]:

determining support for said association rule in said randomized dataset [see section 4.2];

estimating support of said association rule in said original dataset based on said support of said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said randomizing, privacy breaches of said individual transactions are controlled during said mining [see section 4.4].

Claim 13: A computer-implemented method of mining association rules from datasets while maintaining privacy of individual transactions within said datasets through randomization, said method comprising:

creating randomized transactions from an original dataset by [see section 4.1]:

randomly dropping true items from each transaction in said original dataset [see section 4.1], and

randomly inserting false items into each transaction in said original dataset [see section 4.1];

creating a randomized dataset by collecting said randomized transactions [see section 4.1];

collecting said randomized dataset in a database [see section 4.1]; and

mining said database to recover an association rule in said original dataset after said dropping and inserting processes [see sections 4.2-4.5], wherein said mining comprises:

determining support for said association rule in said randomized dataset [see section 4.2];

estimating support of said association rule in said original dataset based on said support for said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said creating of said randomized transactions, privacy breaches of said individual transactions are controlled during said mining [see section 4.4].

Claim 19: A computer program product on a computer-readable medium and tangibly embodying a program of instructions executable by a computer to perform a method of mining association rules from databases while maintaining privacy of individual transactions within said databases through randomization, said method comprising:

randomizing an original dataset to create a randomized dataset [see section 4.1], said randomizing comprising:

randomly dropping true items from each transaction in said original dataset [see section 4.1];

randomly inserting false items into each transaction in said original dataset [see section 4.1];

collecting said randomized dataset in a database [see section 4.1]; and

mining said database to recover an association rule in said original dataset after said dropping and inserting processes [see sections 4.2-4.5], wherein said mining comprises:

determining support for said association rule in said randomized dataset [see section 4.2];

estimating support of said association rule in said original dataset based on said support of said association rule in said randomized dataset [see section 4.3]; and

outputting said association rule if said support of said association rule in said original data set is estimated to be greater than a predetermined minimum [see section 4.5],

wherein, due to said randomizing, privacy breaches of said individual transactions are controlled during said mining [see section 4.4].

[0011] Furthermore, dependent claims 2-6, 8-12, 14-18 and 20-24 are either explicitly described in “Privacy Preserving Mining of Association Rules” (July 2002) or inferred from details contained therein.

[0012] “Privacy Preserving Mining of Association Rules” (July 2002) clearly predates the August 2002 earliest effective prior art date of Rizvi. Additionally, at the August 2002 earliest effective prior art date of Rizvi, the authors of Rizvi had knowledge of the details of the present invention and wrote their paper in light of that knowledge. This is evidenced by the fact that, as mentioned above, the details of the invention as defined by claims 1-24 of the Patent

Application are described in “Privacy Preserving Mining of Association Rules” (July 2002) and further by the fact that Rizvi cites “Privacy Preserving Mining of Association Rules” (July 2002), as a reference, at various places throughout the text of the article.

[0013] We were diligent from the date of conception in reducing the invention to practice and in pursuing, preparing, and filing the Patent Application.

[0014] On May 15, 2003, a patent attorney was instructed to prepare a patent application that eventually became the Patent Application. The Patent Application was eventually prepared and filed on July 21, 2003.

[0015] Finally, the above declarations are made according to the best of my/our recollection upon review of the appropriate documents and notes, and I/we hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the Patent Application or any patent issuing thereon. All statements that are made herein of my/our own knowledge are true and all statements that are made herein based on information and belief are believed to be true.

Alexandre Evfimievski (Date)

Ramakrishnan Srikant (Date)

Rakesh Agrawal (Date)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

We, Alexandre Evfimievski, Ramakrishnan Srikant, and Rakesh Agrawal, the Applicants and joint inventors of the above-referenced invention defined by claims 1-24 and disclosed in U.S. Patent Application Serial No. 10/624,069 hereby declare the following:

[0001] The purpose of this declaration is to prove that we conceived the claimed invention prior to the earliest effective prior art date of **Exhibit A**. Exhibit A is a copy of the following published article cited in the March 5, 2008 rejection of claims 1-24 of the present patent application (herein after referred to as Patent Application) under 35 U.S.C. §102(a): Rizvi, et al., "Maintaining Data Privacy in Association Rule Mining," Proceedings of the 28th VLDB Conference, Hong Kong, China, 12 pages, dated August 2002 (hereinafter referred to as Rizvi).

[0002] The following shows that we conceived our invention prior to the August 2002

earliest effective prior art date of Rizvi, that we were diligent from the date of conception to the date of reduction to practice and that we were further diligent to the date of the filing of the patent application (herein after referred to as Patent Application), which has a filing date of July 21, 2003.

[0003] Proof of the conception of the claimed invention prior to August 2002 and diligence in reducing the invention to practice and filing the Patent Application is demonstrated by the attached **Exhibit B** in conjunction with **Exhibit A**.

[0004] **Exhibit B** is a copy of the following published paper: Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, “Privacy Preserving Mining of Association Rules,” Proc. Of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002, referred to herein as “Privacy Preserving Mining of Association Rules” (July 2002).

[0005] Each of the Applicants of the Patent Application are co-authors on the paper “Privacy Preserving Mining of Association Rules” (July 2002) along with J. Gehrke.

[0006] J. Gehrke was a professor and advisor of A. Evfimievski, during the time period in which the idea for the invention was conceived. Although J. Gehrke is listed as a co-author of “Privacy Preserving Mining of Association Rules” (July 2002), he was not an inventor of the invention defined by claims 1-24 of the Patent Application.

[0007] J. Gehrke has read the Patent Application and has declared that he is not an

inventor of the invention defined by claims 1-24 (**see Exhibit C**). We, the Applicants, also acknowledge that J. Gehrke was not an inventor of the invention defined by claims 1-24 of the Patent Application. Therefore, the portions of “Privacy Preserving Mining of Association Rules” (July 2002), which describe the features of claims 1-24 of the Patent Application, describe the Applicants’ own work and no one else’s.

[0008] “Privacy Preserving Mining of Association Rules” (July 2002) describes the invention defined by claims 1-24. In fact “Privacy Preserving Mining of Association Rules” (July 2002) and, in particular, section 4 of “Privacy Preserving Mining of Association Rules” (July 2002), served as the basis for the specification, drawings and claims of the Patent Application.

[0009] Furthermore, dependent claims 2-6, 8-12, 14-18 and 20-24 are either explicitly described in “Privacy Preserving Mining of Association Rules” (July 2002) or inferred from details contained therein.

[0010] “Privacy Preserving Mining of Association Rules” (July 2002) clearly predates the August 2002 earliest effective prior art date of Rizvi. Additionally, at the August 2002 earliest effective prior art date of Rizvi, the authors of Rizvi had knowledge of the details of the present invention and wrote their paper in light of that knowledge. This is evidenced by the fact that, as mentioned above, the details of the invention as defined by claims 1-24 of the Patent Application are described in “Privacy Preserving Mining of Association Rules” (July 2002) and further by the fact that Rizvi cites “Privacy Preserving Mining of Association Rules” (July

2002), as a reference, at various places throughout the text of the article.

[0011] We were diligent from the date of conception in reducing the invention to practice and in pursuing, preparing, and filing the Patent Application.

[0012] On May 15, 2003, a patent attorney was instructed to prepare a patent application that eventually became the Patent Application. The Patent Application was eventually prepared and filed on July 21, 2003.

[0013] Finally, the above declarations are made according to the best of my/our recollection upon review of the appropriate documents and notes, and I/we hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the Patent Application or any patent issuing thereon. All statements that are made herein of my/our own knowledge are true and all statements that are made herein based on information and belief are believed to be true.

Alexandre Evfimievski (Date)

Ramakrishnan Srikant (Date)

Rakesh Agrawal (Date)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of

Agrawal et al.

Atty. Docket No.: ARC920030034US1

Serial No.: 10/624,069

Group Art Unit: 2161

Filed: July 21, 2003

Examiner: Padmanabhan, Kavita

For: MINING ASSOCIATION RULES OVER PRIVACY PRESERVING DATA

Commissioner of Patents
P.O. BOX 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. §1.131

We, Alexandre Evfimievski, Ramakrishnan Srikant, and Rakesh Agrawal, the Applicants and joint inventors of the above-referenced invention defined by claims 1-24 and disclosed in U.S. Patent Application Serial No. 10/624,069 hereby declare the following:

[0001] The purpose of this declaration is to prove that we conceived the claimed invention prior to the August 2002 date of **Exhibit A**. Exhibit A is a copy of the following published article cited in the March 5, 2008 rejection of claims 1-24 of the present patent application (herein after referred to as Patent Application) under 35 U.S.C. §102(a): Rizvi, et al., “Maintaining Data Privacy in Association Rule Mining,” Proceedings of the 28th VLDB Conference, Hong Kong, China, 12 pages, dated August 2002 (hereinafter referred to as Rizvi).

[0002] The following shows that we conceived our invention prior to the August 2002

date of Rizvi, that we were diligent from the date of conception to the date of reduction to practice and that we were further diligent to the date of the filing of the patent application (herein after referred to as Patent Application), which has a filing date of July 21, 2003.

[0003] Proof of the conception of the claimed invention prior to August 2002 and diligence in reducing the invention to practice and filing the Patent Application is demonstrated by the attached **Exhibit B** in conjunction with **Exhibit A**.

[0004] **Exhibit B** is a copy of the following published paper: Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, “Privacy Preserving Mining of Association Rules,” Proc. Of 8th ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD), July 2002, referred to herein as “Privacy Preserving Mining of Association Rules” (July 2002).

[0005] Each of the Applicants of the Patent Application are co-authors on the paper “Privacy Preserving Mining of Association Rules” (July 2002) along with J. Gehrke.

[0006] J. Gehrke was a professor and advisor of A. Evfimievski, during the time period in which the idea for the invention was conceived. Although J. Gehrke is listed as a co-author of “Privacy Preserving Mining of Association Rules” (July 2002), he was not an inventor of the invention defined by claims 1-24 of the Patent Application.

[0007] J. Gehrke has read the Patent Application and has declared that he is not an inventor of the invention defined by claims 1-24 (**see Exhibit C**). We, the Applicants, also

acknowledge that J. Gehrke was not an inventor of the invention defined by claims 1-24 of the Patent Application. Therefore, the portions of “Privacy Preserving Mining of Association Rules” (July 2002), which describe the features of claims 1-24 of the Patent Application, describe the Applicants’ own work and no one else’s.

[0008] “Privacy Preserving Mining of Association Rules” (July 2002) describes the invention defined by claims 1-24. In fact “Privacy Preserving Mining of Association Rules” (July 2002) and, in particular, section 4 of “Privacy Preserving Mining of Association Rules” (July 2002), served as the basis for the specification, drawings and claims of the Patent Application.

[0009] “Privacy Preserving Mining of Association Rules” (July 2002) clearly predates the August 2002 date of Rizvi, which in fact cites “Privacy Preserving Mining of Association Rules” (July 2002), as a reference, at various places throughout the text of the article.

[0010] We were diligent from the date of conception of our invention in reducing the invention to practice and in pursuing, preparing, and filing the Patent Application.

[0011] On May 15, 2003, a patent attorney was instructed to prepare a patent application that eventually became the Patent Application. The Patent Application was eventually prepared and filed on July 21, 2003.

[0012] Finally, the above declarations are made according to the best of my/our

recollection upon review of the appropriate documents and notes, and I/we hereby acknowledge that willful false statements and the like are punishable by fine or imprisonment, or both (18 U.S.C. 1001) and may jeopardize the validity of the Patent Application or any patent issuing thereon. All statements that are made herein of my/our own knowledge are true and all statements that are made herein based on information and belief are believed to be true.

Alexandre Evfimievski (Date)

Ramakrishnan Srikant (Date)

Rakesh Agrawal (Date)